```
*************************************************************
                                  ***
        /////////      ////   ////  ***  /////////
       IIIIIIIII/     IIII/  IIII/    ***IIIIIIII//
      IIIII/          IIII/  IIII/     IIII/  IIII/
      IIIII/          IIII/  IIII/     IIII/* IIII/
      IIIII/          IIII/  IIII/     IIII/***III/
      IIIII//////  // IIII///IIII/  // IIII///***I/  //
       IIIIIIIII/ II/  IIIIIIIII/  II/  IIIIIIII*** II/
                                                ***
*************************************************************
                  CIRCULO DE USUARIOS DE QL
*************************************************************
```

Fanzine mensual independiente para usuarios de Sinclair QL y compatibles

---

AÑO 2 NUMERO 13 OCTUBRE DE 1989

Estamos en el año 9 Después de Sinclair. Toda Hispania está ocupada por las legiones de PCs. ¿Toda? ¡No! Un puñado de irreductibles QLs resiste todavía y siempre ante el primitivo invasor...

CONTENIDO

INFORMACION SOBRE C.U.Q.

Compilación de colaboraciones: Salvador Merino
Títulos y texto "Información sobre C.U.Q.": Marcos Cruz
Sub-Distribución: Marcos Cruz
Imagen de la portada: SIRVASE USTED MISMO DEL PROGRAMA TORTUGA

El fanzine C.U.Q. es un medio de comunicación e intercambio de ideas y conocimientos entre usuarios de ordenadores Sinclair QL y compatibles. El fanzine se realiza mediante la colaboración desinteresada de todas las personas que lo desean.

¿COMO SE PUEDE RECIBIR EL FANZINE?

El fanzine se envía a quien contribuye a su realización aportando alguna colaboración, o bien a quien lo solicita sin más.

Quienes sólo quieran recibir el fanzine, sin enviar ninguna colaboración para que sea incluida en él, deberán enviar UN disco formateado de 3.5 pulgadas o DOS microdrives formateados, junto con un sobre franquedo y con las señas

escritas para el reenvío. Estos discos o microdrives SIN COLABORACIONES deben enviarse a: Marcos Cruz, Acacias 44, 28023 Madrid

El fanzine de cada mes estará disponible, para ser enviado a quien lo solicite, a finales de la primera semana de dicho mes, con lo cual pueden enviarse las peticiones a comienzos de mes.

¿COMO SE PUEDE COLABORAR EN LA REALIZACION DEL FANZINE?

Quien desee colaborar en la realización del fanzine, debe igualmente enviar UN disco de 3.5 pulgadas o DOS cartuchos de microdrive (aunque sólo uno contenga algo) junto con un sobre franquedao y con la dirección escrita para el reenvío. Estos discos o microdrives CON COLABORACIONES deben enviarse a: Salvador Merino, Ctra. Cádiz (Cerámicas Mary), 29640 FUENGIROLA (Torreblanca del Sol), MALAGA

¿Qué puede enviarse como colaboración? Cualquier cosa: artículos, cartas, opiniones, preguntas, respuestas, ideas, sugerencias... Si bien el nexo de unión entre quienes hacen posible C.U.Q. es el interés y la inquietud por el ordenador Sinclair QL, y por la informática en general, y esa es la temática global del contenido del fanzine, C.U.Q. acepta artículos sobre cualquier tema que pueda interesar a los lectores.

Con todas las colaboraciones recibidas a lo largo de un mes, se confecciona el fanzine correspondiente al mes siguiente. Los discos y microdrives con los boletines se echarán al correo en el último fin de semana del mes.

No deben enviarse más colaboraciones hasta haber recibido de vuelta los discos o microdrives con el fanzine del mes correspondiente.

FORMATO DE LAS COLABORACIONES

Los textos deben ser ficheros _doc del programa Quill, con los siguientes márgenes: izquierdo=1, derecho=80, sangrado=5. Los textos deben ir justificados a la derecha, y no deben emplearse caracteres no castellanos cuyo código ASCII sea mayor de 127. Tampoco deben usarse los corchetes cuadrados, ni las llaves, ni las barras vertical o inclinada hacia la izquierda.

Los artículos deben agruparse temáticamente según las siguientes secciones:

CAR	Cartas abiertas
NOT	Noticias
LIB	Comentarios de libros y otras publicaciones
PRO	Comentarios de programas
HAR	Comentarios sobre hardware, y montajes
BAS	Lenguaje SuperBASIC
FTH	Lenguaje Forth
LEC	Lenguaje C
ASM	Lenguaje ensamblador 68000
COB	Lenguaje COBOL
MAN	Manuales traducidos
ZET	Z88
SPE	Spectrum
OFE	Ofertas, compra-venta, anuncios particulares
OTR	Otros temas de interés, aunque no sean informáticos.

Esta lista podrá ser ampliada o modificada por el recopilador, a sugerencia de los colaboradores.

En el caso de que la colaboración consista en preguntas, o respuestas, deberá igualmente incluirse en una de las secciones indicadas, según el tema de a que se refieran.

Todas los artículos que pertenezcan a una misma sección deben ir en un mismo fichero. El nombre de este fichero tendrá el formato siguiente: SSSxxnnn_doc, donde "SSS" son los tres caracteres correspondientes a la sección de destino del texto, "xx" es el número de fanzine al que se envía el texto y "nnn" son las iniciales del autor.

En el texto, el título de cada artículo deberá ir en mayúsculas, en letra negrita, y a la izquierda de la línea (en el margen de sangrado). Excepto la letra negrita en los títulos de los artículos, no deberá emplearse ningún tipo de letra (subrayada, negrita, subíndice o superíndice) en los textos.

Deberá dejarse siempre UNA SOLA línea en blanco para separar el título de un artículo del artículo anterior, si lo hay. Dentro de cada artículo deberá evitarse por completo el incluir líneas en blanco, tanto para separar párrafos como por cualquier otro motivo, con el objeto de ahorrar espacio en las copias impresas.

Al final de cada cada artículo debe ir el nombre del autor, en la columna 40, seguido en la misma línea por la localidad y la fecha.

Algunas importantes recomendaciones a observar:

-Déjese siempre un espacio detrás de cada signo de puntuación (coma, punto, punto y coma, dos puntos) ¡pero no delante!

-No deben aislarse con espacios los signos de interrogación, admiración, guiones o paréntesis, sino que deben estar unidos a la palabra siguiente -si son signos de apertura- o anterior -sin son de cierre.

-Cuídense tanto la ortografía y las tildes como los errores de "tecleo". Para ello, es recomendable releer lo que se ha escrito.

¿EN QUE FORMATO SE RECIBE EL FANZINE?

El fanzine se recibe en forma de varios ficheros _doc del procesador Quill, al objeto de permitir el acceso al mísmo de los usuarios que no tienen unidades de disco y/o ampliación de memoria.

El nombre de cada fichero indica el número de fanzine al que pertenece y la sección que contiene.

Para componer el fanzine debe cargarse el fichero CUQxxPOR (donde "xx" es el número de fanzine) y seguidamente deben UNIRse al final del texto, uno a uno, los ficheros de las restantes secciones en el orden que indique la columna "Sección" del índice del fanzine. Seguidamente puede alterarse el número de líneas por página, asignarse un pie de página y/o un encabezamiento y, si se desea, añadir cambios de página forzados para evitar que los títulos de secciones o artículos queden al final de alguna página. Finalmente, se ha de grabar el fanzine completo en disco y/o sacar una copia por impresora.

Si, por falta de memoria, no puede componerse todo el fanzine entero, deberá imprimirse por partes, a juicio de cada uno.

Una vez tenemos el fanzine en papel, habrá que rellenar a mano los números de página del índice.

¿PUEDEN INCLUIRSE DIBUJOS EN EL FANZINE?

Normalmente la portada llevará un dibujo (preferiblemente relacionado con alguno de los artículos del fanzine). El dibujo será un fichero de pantalla, en modo 4 u 8. El nombre de estos ficheros debe ser: DBPxxnnn_MODEm, donde "m" es el modo de pantalla en el que están realizados. Como en los textos, "xx" es el número del fanzine de destino, y "nnn" son las iniciales del autor.

El dibujo elegido por el recopilador para ser la portada del mes, se incluirá en el fanzine con el nombre CUQxxDBP_MODEm (donde "xx" es el número de fanzine y "m" indica el modo de pantalla, 4 u 8). Este fichero deberá emplearse para sacar un "hard-copy" sobre el folio de la portada.

La inclusión de dibujos en los artículos aún no es posible, pero están en estudio varios métodos posibles.

¿COMO ENVIAR PROGRAMAS AL FANZINE?

El fanzine C.U.Q. sirve también como intercambio de programas (PROGRAMAS REALIZADOS POR LOS MISMOS COLABORADORES, NO PROGRAMAS COMERCIALES). Estos programas pueden estar realizados en cualquier lenguaje disponible para el QL. En la medida de lo posible, y siempre que su longitud no sea excesiva, los textos fuente de dichos programas deben incluirse en un artículo (en la sección del lenguaje correspondiente) en el que se explique el funcionamiento u otros detalles que el autor considere oportuno. Los listados de programas que se incluyan en el boletín, no deberán tener líneas con más de 80 caracteres. Si el autor prefiere no ceder al fanzine el texto fuente, deberá igualmente escribir las instrucciones de manejo en un pequeño artículo, para su inclusión en el fanzine.

El fanzine C.U.Q. no mantiene una librería de programas como tal, puesto que, por enviarse éstos en el mismo disco del fanzine, no es necesaria. No obstante, si alguien precisa alguno de los programas, puede solicitarlo.

Normas a tener en cuenta para enviar programas al fanzine.

-Póngase a cada fichero una extensión adecuada que indique su tipo, por ejemplo:

_bas si es un programa en SuperBASIC
_exe o _task si es un programa EXECutable
_bin si es código máquina no EXECutable
_txt o _asc si es un fichero de código ASCII
_fth si es un texto fuente es Forth
_scr si es una pantalla, o un fichero fuente del Forth de Computer One
...etcétera

-Póngase a todos los ficheros de un mismo programa un mismo nombre inicial, que no coincida con alguna de las referencias de los programas ya existentes. Si entre los ficheros de un programa existe un fichero _doc del Quill, o de otro paquete de Psion, el nombre inicial común a todos los ficheros no podrá, por tanto, ser más largo de 8 caracteres.

-No deben emplearse en los nombres de los ficheros caracteres que no sean aceptados directamente, es decir, que no sea necesario encerrar el nombre entre

comillas para manipular el fichero.
-Deben escribirse instrucciones detalladas del manejo de los programas, bien dentro del mismo programa, o en un artículo que aparezca en el fanzine, o bien en un fichero _doc.
Los programas deben enviarse, como las colaboraciones escritas, a: Salvador Merino, Carretera de Cádiz (Cerámicas Mary), 29640 FUENGIROLA (Torreblanca del Sol), Málaga.

EDITORIAL

En el momento de escribir este editorial estamos a mediados de septiembre (un fanzine CUQ se prepara varias semanas antes del mes que va en la portada).
Los veranos siempre han sido flojos en novedades y noticias, y este no es una excepción. Aún asi, creo que los Dioses no nos han abandonado todavía.
El futuro del QL, comercialmente hablando, está por los suelos (he descubierto la piedra). La única esperanza que nos queda es el THOR, el cual tiene un precio que es para tirarse de los pelos. Pero si el proyecto THOR siguiese adelante con unas ventas minimas (500 máquinas mes) y saliendo nuevos modelos basados en los motorolas grandes (MC 68020 para arriba), podria continuar la oferta de software para QL y THOR, porque a las casas de software solamente le interesan máquinas en producción (p.e.: el viejo Apple II aún se fabrica) y no el número de máquinas vendidas (p.e.: en el mundo PC las ventas de originales son ridiculas comparadas con el parque de máquinas instaladas).
Dejando de lado el futuro del THOR, he de advertir que nuestro futuro como Circulo de Usuarios es muy bueno. El próximo número está garantizado, porque hay un material de engorde en reserva, pero necesito más cartas abiertas, articulos de todo tipo y pequeños programitas.
Actualmente somos suficientes para que con solamente escribir 10 páginas cada uno en 1990, podamos llenar entre todos los 12 números de CUQ. Eso significa escribir un par de lineas diarias, pero mucho me temo que, como siempre, la gran labor CUQ sea escrita por solamente cuatro gatos mal contados cuyos nombres conocemos todos.
Si repasais todo el material CUQ editado hasta hoy vereis que no cabe en cuatro discos 3.5" de 720K. Mucha gente (socios de QLave) ya me dicen que en solamente un año hemos editado más material que QLave en dos años. Yo nunca he pedido dinero para cubrir gastos, solamente he pedido BYTEs.
Me gustaria conocer por antemano todo lo que le va a ocurrir a CUQ en 1990, pero nuestro futuro se encuentra en vuestra manos, quizás el nombre de nuestro Club debia ser Grupo Interesado por el QL (GIQ).
S. Merino, 12/9/1989

NOTICIAS

¿FIG ESPAÑA?

Hace unos dias he recibido una carta de un miembro nuevo en FIG este año, Borja Marcos, el cual es usuario de PC, BBC MICRO y PSION ORGANISER, pero ha pedido información a FIG para formar un grupo local en España.
Por lo visto en España solamente hay actualmente 10 miembros del FIG: 3 en Barcelona, 2 en Madrid, 1 en Pamplona, 1 en Sevilla, 1 en Zaragoza, 1 en Málaga y 1 en Bilbao.
En FORTH DIMENSIONS VOLUME XI, NUMBER 2 JULY/AUGUST 1989, Marcos Cruz

ha conseguido publicar su articulo EXPERT SYSTEM TOOLKIT, el cual corresponde al articulo SISIFOrth publicado en Fanzine CUQ número 2. Gracias a esto, el miembro FIG de Zaragoza, que es usuario de QL, se ha puesto en contacto con Marcos y se apunta a lo de CUQ.
Actualmente CUQ posee el 40% de los socios de FIG en España e intercambia correspondecia con el 50%. Solamente se necesitan 5 miembros para formar el Grupo Local FIG en España, creo que somos suficientes.
Borja Marcos tiene en proyecto implementar el fig-forth en su Psion Organiser, ello me ha dado la idea de que yo podria escribir mi propia versión del Forth para el Z88 si consigo suficiente documentación de FIG. Existe un libro, THREAD INTERPRETIVE LANGUAGES, que explica paso a paso el desarrollo de un no standard Z-80 FORTH. También existe un listado en assembler Z80 del fig-FORTH, pero no creo que eso pueda correr bajo OZ sin hacer ninguna modificación (Aunque ¿quién sabe?).
Salvador Merino, 12/9/1989

Hoy he recibido una carta de Borja Marcos y una hora después una llamada telefónica suya. Las nuevas noticias son que el número de socios de FIG a aumentado a 11 con un nuevo miembro en Navarra usuario de Jupiter Ace, el cual fue quien convenció a Borja mostrandole la potencia del FORTH.
El sistema de comunicación que se le ha ocurrido es el siguiente:
Dejaremos en Informática Algorta tres noches a la semana (de 10 de la noche a 9 ó 9:30 de la mañana) un PC encendido en modo host como un pequeño BBS; En principio se dedicaría a programas, información, y a organizar un "lugar" donde llevar a cabo debates, etc. simplemente dejando ficheros de texto; al mismo tiempo, estoy pensando en un sistema sencillito de correo electrónico. Lo que sí habrá que tener en cuenta es que nadie deberá "prestar" su password; si alguien quiere registrarse que se ponga en contacto conmigo, ya que por sus características no la podemos hacer abierta al público, al no disponer de un software en condiciones para BBS's y disponer sólo de una partición de 5MB que destinaré al efecto. Además la forma de acceso será directo al DOS desde el terminal, es decir, algo que no se puede permitir a cualquiera.
De este modo se pueden mantener teleconferencias de una forma cómoda y sencilla, y coger o dejar programas, comentarios, etc.
Para los que no tengan modem podemos utilizar el sistema del diskette que me comentabas con el sobre y el franqueo de respuesta, con lo que todos podremos acceder. Además aqui tenemos también discos de 3.5", o sea, no hay excusa para decir que no se puede. En los discos se podrían mandar también los ficheros de las teleconferencias actualizados, para que todo el mundo pueda participar. Con el sistema del modem, todo coste es la llamada, pero si, por ejemplo, para participar en una teleconferencia se llama se cogen los textos, se desconecta, se redacta un texto y se vuelve a llamar con el texto ya escrito, el coste de la llamada sería el minimo. Además, se pueden pasar los ficheros en formato.ARC. (Aunque a ver cómo lo hacemos en un QL) De todas formas, desde el terminal se pueden perfectamente empaquetar o desempaquetar ficheros llamando al programa que lo hace. También podríamos prepararlos en formato Z88.

También ha confesado que su PC es de la empresa y él odia el PC.
S. Merino, 13/9/1989.

QL, ¿UNA PIEZA DE MUSEO?

El QL se encuentra expuesto en el museo de Ciencia y Tecnologia de Manchester (el museo industrial más grande de Europa) desde su muerte oficial en marzo de 1986 con una copia de Sinclair QL World de esa misma fecha, un Sinclair Vision QL monitor, y una historia de la máquina (que termina en Marzo 1986).
El Times escribió un corto articulo en navidades del año pasado comentando que el QL se encontraba en un Museo.
S. Merino

TRADUCTOR DE TEXTOS PARA QL/Z88

Cuando uno lee códigos ASCII Standard, uno cree que todos los ordenadores poseen el mismo juego de caracteres ASCII, pero la realidad es muy distinta. Aún siendo del mismo fabricante, existen unos códigos ASCII para representar los caracteres no Ingleses, los cuales en su época no fueron Standarizados porque los ordenadores eran todos Anglosajones. Esos códigos normalmente ocupan valores por encima de 127 a 255, pero también hay diferencias en los códigos de control (por debajo del valor 32).

Como detalle curioso, habria que poner al engendro negro que tenemos debajo de nuestros dedos, el QL. No sé si alguno no se ha enterado, pero todos los QLs poseen de fábrica los mismos caracteres en su juego ASCII, pero, y valga la redundancia, los caracteres extranjeros poseen códigos diferentes según la versión del QL. Eso quiere decir que no solamente se diferencian un QL Inglés, Alemán, Frances, Español,... en que los teclados son diferentes, sino en que tampoco coinciden los valores ASCII de los caracteres que solamente se dan en determinados idiomas. Y señores estamos hablando del QL (creo que en el THOR XVI se ha solucionado la papeleta definitivamente).

Todo lo anterior es una introducción sobre el tema, porque he recibido el QZ/QL to Z88 File Transfer, y no hay que ser tonto para darse cuenta de que los Ingleses, como siempre, no han tenido en cuenta el detalle expuesto en la traducción. El último código ASCII traducido es el retorno de carro (QL usa chr$(10) y Z88 usa chr$(13)) siempre y cuando lo deseemos, porque es opcional.

Y ahora más de un cateto diria, ¿Anda y ahora que hacemos?. Si es un usuario de PC seguro que le sacan la pasta bien sacada si su PC LINK no hace la traducción. En nuestro caso, somos programadores por Hobby o por dinero (da lo mismo, la cuestión es que sabemos programar), hacer un programa traductor para el QL es cosa de niños.

Tenia pensado hacer una versión de pruebas sin ningún tipo de diseño de pantalla, pero enamorado del diseño que hizo José Carlos de Prada en su programa Traductor IBM/QL, decidí hacer todas las modificaciones en ese programa para transformarlo en TRADUCTOR QL/Z88. La única pega era la velocidad, pues traducir el comentario QZ desde RAMDISK supuso más de 3 minutos (perdí la cuenta).

Naturalmente me decidí por compilarlo con QLIBERATOR. Segunda Pega, José Carlos habia utilizado el WHEN ERRor del Toolkit II que se puede compilar, pero no es posible compilar RETRY, pues en un programa compilado no existen los números de linea. Asi que eliminé todo el WHEN ERRor para compilarlo rapidamente y la traducción del comentario QZ fue esta vez de solamente 40 segundos en RAMDISK.

No contento aún con el cambio de velocidad, me dió por compilarlo con TURBO v2.0 (poco me falto para volverme loco, os lo aseguro). Si en el Qliberator compilarlo fue coser y cantar a la primera, en el Turbo compilarlo me costó varias horas y no se cuentas compilaciones. Lo primero, ¡Viva la Pepa!, advertencias, ¡Hay que dimensionar las variables de cadena! (P.e.: DIM disp$(100) por defecto, no me he entretenido en contar los bytes necesarios). Lo segundo se compila sin errores y advertencias, pero.... El programa hace un Halted cada dos por tres en dos lugares similares. ¿Qué pasa? Que el Turbo se me ha puesto chulo a mi. La respuesta se encuentra en el manual, o mejor dicho enciclopedia Turbo, el programa Traductor usa el TOOLKIT II y aunque Digital Precision dice que el Turbo v2.0 es totalmente compatible con Toolkit II, sigue siendo compatible a medias. Los comandos PUT, GET, BPUT y BGET no corren bien compilados con TURBO (en el manual de la v1.09 lo advierte). Después de casi reescribir todo el programa, conseguí una versión digna de ser comercializada. Y llegó la hora de la verdad, traducir el comentario QZ cuesta desde RAMDISK 35 segundos. ¡Solamente 5 segundos de diferencia con el Qliberator! ¡Aaarghh! Y para eso me he molestado yo en reescribir todo el programa para poderlo compilar con TURBO, solamente existe menos de un 20% de incremento de velocidad.

A continuación os voy a poner el volcado de la versión TURBO. La versión compilada Turbo (con runtime) y una versión Qliberator en intérprete y compilado (ambos necesitan TOOLKIT II) seran enviadas casi

con toda seguridad junto a este número. Quizás me decida, si no me da por otra cosa, por escribir la versión en Assembler, la cual es una tarea de chinos y no hay garantias de un aumento considerable de velocidad (Más problemas, ¡Oh no! ¡¡Socorro!! HELP!)

```
10 REMark ***************** Traducción Z88-QL ***********************
20 REMark ******************** MERINO-SOFT **************************
30 REMark ******************** JULIO 1.989 **************************
40 DIM disp$(100): DIM fuente$(100): DIM destino$(100)
50 OPEN #3,con_400x200a56x23
60 OPEN #4,scr_360x180a76x33:CSIZE#4,1,0
70 PAPER#3,0:BORDER#3,1,255
80 REPeat bucle
90      CLS#3
100      OVER#3,1:CSIZE#3,2,1
110      FOR i=0 TO 7
120              CURSOR#3,138-i,30-i:INK#3,i
130             PRINT #3,"TRADUCTOR"
140     END FOR i
150     OVER #3,0:CSIZE #3,2,0:INK#3,2
160     AT#3,7,9:PRINT #3,"1   QL  ?  Z88"
170     AT #3,9,9:PRINT #3,"2   Z88 ?   QL"
180     AT#3,11,9:PRINT #3,"3   Directorio"
190     AT#3,13,9:PRINT #3,"4   Ver fichero"
200     AT#3,16,9:PRINT #3,"ESC  =  Salida"
210 AT#3,0,0:CURSOR_ON#3
220     op=CODE(INKEY$(#3,-1))
230 CURSOR_OFF#3
240     SELect ON op
250         =49
260             QZ
270         =50
280             ZQ
290         =51
300            unidad
310         =52
320             fichero
330         =27
340             EXIT bucle
350         =REMAINDER
360             BEEP 200,30
370     END SELect
380 END REPeat bucle
390 CLOSE#3:CLOSE#4
400 DEFine PROCedure QZ
410     CLS#3
420     OVER#3,1:CSIZE#3,2,1
430     FOR i=0 TO 7
440             CURSOR#3,128+i,30-i:INK#3,i
450             PRINT #3,"QL  ?  Z88"
460     END FOR i
470     OVER#3,0:INK#3,2:CSIZE#3,2,0
480     AT#3,7,3:PRINT #3,"Fuente: "
490     AT#3,8,3:PRINT #3,"  (Ej.: FLP1_FichQL_TEX)"
500 REPeat err3
510 AT#3,7,11:PRINT #3,FILL$(" ",22):AT#3,7,11:CURSOR_ON#3:INPUT#3,fuente$:
    CURSOR_OFF#3
520 IF DEVICE_STATUS(fuente$)=-8 OR DEVICE_STATUS(fuente$)=-20 THEN EXIT err3
530 BEEP 200,30
540 END REPeat err3
550 OPEN_IN#6,fuente$
560     CLS#3
570     OVER#3,1:CSIZE#3,2,1
580     FOR i=0 TO 7
590             CURSOR#3,128+i,30-i:INK#3,i
600             PRINT #3,"QL  ?  Z88"
610     END FOR i
620     OVER#3,0:INK#3,2:CSIZE#3,2,0
630     AT#3,7,3:PRINT #3,"Destino: "
640     AT#3,8,3:PRINT #3,"   (Ej.: FLP2_FichQL_TEX)"
```

```
650 REPeat err4
660     AT#3,7,12:PRINT #3,FILL$(" ",21):AT#3,7,12:CURSOR_ON#3:INPUT#3,destino$:
    CURSOR_OFF#3
670 IF DEVICE_STATUS(destino$)>=0 THEN EXIT err4
680 BEEP 200,30
690 END REPeat err4
700 OPEN_NEW#5,destino$
710     CLS#3:AT#3,12,3:PRINT #3,"Traduciendo QL  ?  IBM ..."
720     REPeat traQZ
730             IF EOF(#6) THEN EXIT traQZ
740             a%=CODE(INKEY$(#6))
750             selector=a%
760             SELect ON selector
770                 =10
780                   PRINT#5,CHR$(13);
790                 =179
800                   PRINT#5,CHR$(161);
810                 =96
820                   PRINT#5,CHR$(163);
830                 =180
840                   PRINT#5,CHR$(191);
850                 =168
860                   PRINT#5,CHR$(199);
870                 =141
880                   PRINT#5,CHR$(224);
890                 =169
900                   PRINT#5,CHR$(209);
910                 =140
920                   PRINT#5,CHR$(225);
930                 =130
940                   PRINT#5,CHR$(226);
950                 =136
960                   PRINT#5,CHR$(231);
970                 =144
980                   PRINT#5,CHR$(232);
990                 =131
1000                  PRINT#5,CHR$(233);
1010                =145
1020                  PRINT#5,CHR$(234);
1030                =148
1040                  PRINT#5,CHR$(236);
1050                =147
1060                  PRINT#5,CHR$(237);
1070                =149
1080                  PRINT#5,CHR$(238);
1090                =146
1100                  PRINT#5,CHR$(239);
1110                =137
1120                  PRINT#5,CHR$(241);
1130                =151
1140                  PRINT#5,CHR$(242);
1150                =150
1160                  PRINT#5,CHR$(243);
1170                =152
1180                  PRINT#5,CHR$(244);
1190                =132
1200                  PRINT#5,CHR$(246);
1210                =154
1220                  PRINT#5,CHR$(249);
1230                =153
1240                  PRINT#5,CHR$(250);
1250                =155
1260                  PRINT#5,CHR$(251);
1270                =135
1280                  PRINT#5,CHR$(252);
1290               =REMAINDER
1300                   PRINT#5,CHR$(a%);
1310            END SELect
1320    END REPeat traQZ
1340    CLOSE#6
```

```
1350    CLOSE#5
1360    CLS#3
1370    AT#3,12,6:PRINT#3,"Proceso concluido"
1380    PAUSE 25
1390 END DEFine
1400 DEFine PROCedure ZQ
1410    CLS#3
1420    OVER#3,1:CSIZE#3,2,1
1430    FOR i=0 TO 7
1440            INK#3,i:CURSOR#3,128+i,30-i
1450            PRINT #3,"Z88  ?  QL"
1460    END FOR i
1470    OVER#3,0:INK#3,2:CSIZE#3,2,0
1480    AT#3,7,3:PRINT #3,"Fuente:"
1490    AT#3,8,3:PRINT #3,"  (Ej.: FLP1_FichZ88_TEX)"
1500 REPeat err5
1510 AT#3,7,11:PRINT #3,FILL$(" ",22):AT#3,7,11:CURSOR_ON#3:INPUT#3,fuente$:
    CURSOR_OFF#3
1520 IF DEVICE_STATUS(fuente$)=-8 OR DEVICE_STATUS(fuente$)=-20 THEN EXIT err5
1530 BEEP 200, 30
1540 END REPeat err5
1550 OPEN_IN#6,fuente$
1560    CLS#3
1570    OVER#3,1:CSIZE#3,2,1
1580    FOR i=0 TO 7
1590            INK#3,i:CURSOR#3,128+i,30-i
1600            PRINT #3,"Z88  ?  QL"
1610    END FOR i
1620    OVER#3,0:INK#3,2:CSIZE#3,2,0
1630    AT#3,7,3:PRINT #3,"Destino:"
1640    AT#3,8,3:PRINT #3,"   (Ej.: FLP2_FichQL_TEX)"
1650 REPeat err6
1660 AT#3,7,12:PRINT #3,FILL$(" ",21):AT#3,7,12:CURSOR_ON#3:INPUT#3,destino$:
    CURSOR_OFF#3
1670 IF DEVICE_STATUS(destino$)>=0 THEN EXIT err6
1680 BEEP 200,30
1690 END REPeat err6
1700 OPEN_NEW#5,destino$
1710    CLS#3:AT#3,12,3:PRINT #3,"Traduciendo Z88  ?  QL ..."
1720    REPeat traZQ
1730            IF EOF(#6) THEN EXIT traZQ
1740         a%=CODE(INKEY$(#6))
1750            selector=a%
1760            SELect ON selector
1770                =13
1780                   PRINT#5,CHR$(10);
1790                =161
1800                  PRINT#5,CHR$(179);
1810                =163
1820                  PRINT#5,CHR$(96);
1830                =191
1840                  PRINT#5,CHR$(180);
1850                =199
1860                  PRINT#5,CHR$(168);
1870                =209
1880                  PRINT#5,CHR$(169);
1890                =224
1900                  PRINT#5,CHR$(141);
1910                =225
1920                  PRINT#5,CHR$(140);
1930                =226
1940                  PRINT#5,CHR$(130);
1950                =231
1960                  PRINT#5,CHR$(136);
1970                =232
1980                  PRINT#5,CHR$(144);
1990                =233
2000                  PRINT#5,CHR$(131);
2010                =234
2020                  PRINT#5,CHR$(145);
```

```
2030                 =236
2040                   PRINT#5,CHR$(148);
2050                 =237
2060                   PRINT#5,CHR$(147);
2070                 =238
2080                   PRINT#5,CHR$(149);
2090                 =239
2100                   PRINT#5,CHR$(146);
2110                 =241
2120                   PRINT#5,CHR$(137);
2130                 =242
2140                   PRINT#5,CHR$(151);
2150                 =243
2160                   PRINT#5,CHR$(150);
2170                 =244
2180                   PRINT#5,CHR$(152);
2190                 =246
2200                   PRINT#5,CHR$(132);
2210                 =249
2220                   PRINT#5,CHR$(154);
2230                 =250
2240                   PRINT#5,CHR$(153);
2250                 =251
2260                   PRINT#5,CHR$(155);
2270                 =252
2280                   PRINT#5,CHR$(135);
2290                 =REMAINDER
2300                   PRINT#5,CHR$(a%);
2310             END SELect
2320    END REPeat traZQ
2330    CLOSE#6
2340    CLOSE#5
2350    CLS#3
2360    AT#3,12,6:PRINT#3,"Proceso concluido"
2370    PAUSE 25
2380 END DEFine
2390 DEFine PROCedure unidad
2400              REPeat err1
2410 CLS#3:AT#3,16,3:CURSOR_ON#3 :INPUT#3,"Unidad (Ej.: FLP1_): ";disp$:CURSOR_OFF#
    3:CLS#3:CLS#4
2420 IF DEVICE_STATUS(disp$)>=0 THEN EXIT err1
2430 BEEP 200,30
2440 END REPeat err1
2450 DIR#4,disp$
2460            PAUSE
2470 END DEFine unidad
2480 DEFine PROCedure fichero
2490            REPeat  err2
2500 CLS#3:AT#3,16,3:CURSOR_ON#3:INPUT #3,"Fichero: ";disp$:CURSOR_OFF#3:CLS#3
2510 IF DEVICE_STATUS(disp$)=-8 OR DEVICE_STATUS(disp$)=-20 THEN EXIT err2
2520 BEEP 200,30
2530 END REPeat err2
2540 VIEW#4,disp$
2550            PAUSE
2560 END DEFine fichero
```

La versión Qliberator con manejo de errores suministrada junto a este número ha sido terminada de confeccionar en mi Z88, lo que significa que puede usarse sin problemas para escribir programas para el QL (creo que voy a escribir más programas en el Z88 para el QL que para el propio Z88).

S. Merino, Fuengirola, 17/6/1989

TORTUGAS, CARACOLES, & MEJILLONES

De verdad ha llovido mucho desde que empeze a escribir este programilla en mayo del 85. Por entonces yo acababa de comprarme el QL, no tenia ni un solo periferico, no tenia ni idea de programacion, ni idea de matematicas y lo que era peor ni idea de como descifrar el QL User Guide que para postre estaba en chino. Sin embargo, tampoco hace falta nada de todo eso para crear algo curioso en BASIC; realmente el TORTUGAS es una prueba de la potencia que puede tener un

ordenador aun cuando no se sabe usar.

La idea original se me ocurrio un dia paseando por la seccion de -informatica de El Corte Ingles. Alli habia varias de esas demostraciones -graficas que tantos curiosos suelen atraer en los grandes almacenes; asi que decidi programar algo similar en mi maquina solo para divertirme. Aquel mismo dia habia comprado el Manual de Referencia del QL de RA-MA y los graficos en Logo que describe fueron el punto de partida. Escribi varias docenas de subrutinas cortas, luego me aburri y lo deje. A decir verdad es sorprendente que hayan llegado hasta aqui, muchas de ellas estaban en papel y se perdieron despues de dos mudanzas sucesivas y el resto estuvo a punto de morir a manos de un formateo que fallo milagrosamente.

El tortuga lleva doce de estas subrutinas. La mayoria son viejas aunque he añadido algunas nuevas para hacer mas interesante el programa. Las subrutinas 1,2 y 9 (VIDRIERA, TABLERO y BOLICHE) las retoque con el EYE-Q para darles un poco mas de color. Hay tres pantallas, de loas tres procedures, grabadas con la opcion de compresion del EYE-Q para no consumir todo el espacio de los lectores que usan cartuchos pero el propio programa las descomprime y su pueden regrabar como una pantalla normal. La ultima subrutina -la numero 12- la escribi la semana pasada en LISP asi que todo lo que hay de ella es una pantalla procedente del EYE-Q. Esta ultima pantalla es la unica que debe necesariamente cargarse en modo 4. Aunque concebi el programa para correr con 8 colores la mayor parte de los dibujos se ven perfectamente en alta resolucion e incluso algunos hasta mejor que usando los 8 colores.

Existen, ademas de estos doce dibujos, otras funciones adicionales; una para dibujar poligonos regulares y otras cinco que dibujan los poliedros regulares.

Para dibujar un poligono la forma es: POLIGONO lados,longuitud_del_lado.

Esto dibujara un poligono en la posicion actual de la tortuga.

En cuanto a los poliedros, funcionan de forma un poco diferente. En primer lugar hay que especificar el nombre del poliedro que se quiere dibujar.

Por ejemplo: TETRAHEDRO 50

Dibujara un tetrahedro de 50 pixels de lado en la posicion actual de la tortuga.

Las funciones HEXAHEDRO y OCTAHEDRO funcionan igual que la anterior.

Pero en el caso del dodecahedro y el icosahedro la cosa se complica un poco mas. Al principio intente diseñarlas igual que las anteriores, basandome en funciones trigonometricas, pero era para alucinar albondigas, de modo que simplemente las hice a ojo en papel milimetrado y luego las traspase a la pantalla usando un semicirculo.

La forma de usarlas es especificar la escala a la que se desean las figuras. Una escala de 100 creara poliedros de unos 20 pixels de lado, una de 50 de 40 pixels, etc. Pero, ademas, hay que especificar el punto donde deseamos que se dibuje la figura en puntos expresados en una escala de 100. Es decir:

DODECAHEDRO 200,70,50

Dibujara un dodecahedro de unos 10 pixels de lado mas o menos en el centro de la pantalla (coordenada x=70, coordenada y=50).

Todas las rutinas que dibujan figuras geometricas son de proposito general, o sea que se pueden utilizar en cualquier otra parte del programa. Por ejemplo prueba esto:

```
FOR n=3 to 40
  POINT 70,52
  octahedro n
NEXT n
```

Solo son 53 bytes pero como ya he dicho tampoco hace falta nada mas para que el resultado sea satisfactorio.

¡¡Ah!!!, y un truco, cualquiera de estos bucles da unos resultados -esplendidos en la impresora. Repasados con un Rotring grueso y coloreados a mano haran tirarse de los pelos a cualquier profesor de dibujo tecnico.

Sergio Montoro
Septiembre 89

COBOL

IECsoft COBOL VS 1.0 (tercera entrega)

Como veis esto pertene al més de Febrero... Y ya es Junio, a este ritmo, esto puede ser la Catedral de Colonia... bueno, vamo a darle un buen empellón. Ele!

```
/* Compilador COBOL V$ 1.00  Valladolid 22 Abril 1988
         © IECSoft 1988
*/
#include <flp1_stdio.h>
#define comillas 34
#define punto 1
unsigned char buffer[80]; /* Buffer para nombres de canal y errores */
int  *chin,*chout,*chwrk,*cherr,errores=0,avisos=0,debuger=0;
char nout[34]; /*nombre del fichero de salida*/
unsigned char cadenaf[320],cadena1[320],cadena2[80]; /* Cadenas de entrada del
    Parser */
char *word[50]; /* matriz de punteros a palabras en la cadena1 */
char margen[50];/* matriz de margen (A o B) de las palabras */
extern unsigned char tabla_ch[];/* Tabla de conversión a mayúsculas */
int  tot_pal;/* total de "palabras" en cadena1 */
extern int puntcola;

/* ************************************************************
          P R O G R A M A   P R I N C I P A L
   ************************************************************ */
main()
{
int sw=1,i=0;
unsigned char tecla;
         crea_nul(); /* conectamos un dispositivo de directorio nulo
                        que se utilizará en caso de error para desviar
                        las salidas al fichero objeto */
         selwindow(stdout);
         cls(0);
         puts("\n   IDIOMA? 1-ENGLISH  2-ESPAÑOL");
         while(sw)
              {
                   tecla=inkey(-1,stdin);
                   switch (tecla)
                   {
                            case '1':
                                     cherr=fopen("flp2_english_msg","r");
                                     sw=0;break;
                            case '2':
                                     cherr=fopen("flp2_español_msg","r");
                                     sw=0;break;
                            default:
                                     continue;
                   }
         }
         csize(1,1);
         puts("\nIECSoft QLCOBOL V$1.0 (C)1988 Nacho Enrique");
         csize(0,0);
         input(buffer,80,cherr);
         puts(buffer);
         input(buffer,80,cherr);
         printf(buffer);fflush(stdout);
         input(cadena1+7,320,stdin);
         convertir();
         chin=fopen(word[0],"r");
         chout=fopen(word[1],"w"); strcpy(nout,word[1]);
         chwrk=fopen(word[2],"w");
         compilar();
}

/* ************************************************************
          FUNCIONES PARA GESTION DE ERRORES
   ************************************************************ */
```

```
mal(ercod)int ercod;
{
          if(!errores++)
                    {
          fclose(chout);delete(nout);
         chout=fopen("nul1_out","w");
                    }
          m_err();
          fallo(ercod);
}
avisar(ercod) int ercod;
{
          avisos++;
          m_err();
          fallo(ercod);
}
m_err()
{
int i,j;
          puts(cadenaf);
          puts("");
          tab(word[puntcola]-cadena1); puts("|");
}
fallo(ercod) int ercod;
{
          ercod<<=6;
          lseek(cherr,ercod,0);
          input(buffer,80,cherr);
          puts(buffer);
          putchar(7);
}


/*  **************************************************************
                FINALIZACION DEL COMPILADOR
    ************************************************************** */

final()
{
         fclose(chin);fclose(chout);fclose(chwrk);
         lseek(cherr,64,0);
         input(buffer,80,cherr);
         printf(buffer,errores,avisos);
         puts(" ");
         input(buffer,80,cherr);
         if(errores) {
                      puts(buffer);
                     }
          fallo(2);
          inkey(-1,stdin);
          desc_nul(); /* Desconectamos la unidad nula */
}
#include <flp2_qlcobol.h>
int puntcola;
/* ***************************************************************
          FUNCION DE ENTRADA DE CADENA DE FICHERO FUENTE
   lee una cadena del fichero de entrada y si no es fin de fichero
   la desmenuza en palabras. En caso de ser fin de fichero y el
   codigo de division 1 aborta el programa con mensaje de error
   fin de fichero fuente inexperado. La función retorna o si no
   hay error y 1 si se llegó al fin de fichero
   ************************************************************** */
entrada(codigo) int codigo;
{
int swe;
swe=leein();
if(swe)
          {
          if(codigo) mal(3);
          final();
```

```
            abort(0);
            }
    convertir();
    puntcola=0;
    return swe;
    }
    palabra(codigo) int codigo;
    {
    int i;
    if(++puntcola>tot_pal)
            {
            if(entrada(codigo)) return 1;
            }
    return puntcola;
    }
    newline(codigo) int codigo;
    {
    if(puntcola)
            {
            if (puntcola!=tot_pal)avisar(8);
            return entrada(codigo);
            }
    return 0;
    }
    /* ************************************************************
                    FUNCION CLAVE
            Averigua si la palabra actual es la que se espera
            si lo es devuelve 1, si no 0. Comprueba también si
            el margen es correcto, de no serlo genera los
            errores correspondientes
       ************************************************************ */
    clave(cadena,lado) char lado, *cadena;
    {
    if(strcmp(cadena,word[puntcola])) return 0;
    margina(lado,puntcola);
    return 1;
    }
    clave2(cadena,lado,pos) char lado, *cadena;int pos;
    {
    if(strcmp(cadena,word[pos])) return 0;
    margina(lado,pos);
    return 1;
    }
    margina(lado,pos) char lado; int pos;
    {
    switch(lado)
            {
            case 1: if(margen[pos]!=1) mal(18);break;
            case 'A':if(margen[pos]!='A') mal(17);break;
            case 'B':if(margen[pos]!='B') mal(15);
            }
    }
    /* ************************************************************
           B U C L E   D E   C O M P I L A C I O N
       ************************************************************ */
    compilar()
    {
    int swc;
            puntcola=tot_pal=0;
            if(ident())
            {
            if(clave("DATA",'A'))  mal(11);
            else if(clave("PROCEDURE",'A'))
                    {swc=3;avisar(13);mal(11);mal(12); }
                    else {mal(14); swc=envir();}
            }
            else swc=envir();
            if(swc==2) swc=datad();
            if (swc==3) proced();
     }
```

```
/* ************************************************************
          IDENTIFICATION DIVISION
   ************************************************************ */
ident()
{
int i,swi;
short largo,salto;
   entrada(1);
   if(clave2("DIVISION",'B',1))
   {
   if(!clave("IDENTIFICATION",'A'))
          {
          avisar(6);
          if(clave("ENVIRONMENT",'A')) return 0;
          if(clave("DATA",'A')
          ||clave("PROCEDURE",'A')) {avisar(9); return 2;}
          mal(19);
          }
    salto_p3(1);
   }
   else {avisar(6);mal(19);}
   if(clave("PROGRAM-ID",'A'))
       {
        clave2("",1,palabra(1));

/* Ponemos la cabecera standard de QDOS en el fichero objeto */

      largo=strlen(word[palabra(1)]);/*largo de el nombre del programa */
      salto=8+largo;
      MBRA(salto);
      puthex("00004afb");
      putshort(&largo);
      fprintf(chout,word[puntcola]);
          coment();
      }
      else mal(7);
      newline(1);
      while(strcmp("DIVISION",word[1]))
      {
   if(!item_id("DATE-WRITTEN")) continue;
   if(!item_id("DATE-COMPILED")) continue;
   if(!item_id("INSTALLATION")) continue;
   if(!item_id("SECURITY")) continue;
   if(!item_id("REMARK"))continue;
      mal(10);
      palabra(1);
      }

   if(clave2("ENVIRONMENT",'A',0)) return 0;
 avisar(9); return 2;
}
item_id(cadena) char *cadena ;
{
      if(!clave(cadena,'A')) return 1;
      clave2("",1,palabra(1));
      return coment();
}
/* ****************************************************************
                    FUNCION DE BUSQUEDA de PUNTO
   devuelve la posición en la cadena en lectura en la que se
   haya el siguiente punto, lee las cadenas que hagan falta para
   encontrarlo
   **************************************************************** */
coment()
{
   while(!clave("",1)) palabra(1);
   return newline(1);
}
salto_p3(i) int i;
{
```

```
    palabra(i);
    clave2("",1,palabra(i));
   newline(i);
}
int base_nul;

/* ******************************************************
         CREACION DEL DISPOSITIVO NULO
   ****************************************************** */

crea_nul()
{
          base_nul=alchp(150);
          base_nul;
/* en A0 tenemos base_nul */
#asm
          movem.l   d0-d7/a0-a6,-(a7)
          lea       $1c(a0),a3
          lea       CN.nulo(PC),a2
          move.l    a2,(a3)+   entrada salida
          lea       CN.close(PC),a1
          move.l    a2,(a3)+   open
          move.l    a1,(a3)+   close
          move.l    a2,(a3)+   dependencia forzada
          addq.l    #8,a3
          move.l    a2,(a3)+
          move.l    #90,(a3)+
          move.w    #3,(a3)+
          move.l    #'NUL ',(a3)
          lea       $18(a0),a0
          moveq     #$22,d0   MT.LDD
          trap      #1
          movem.l   (a7)+,d0-d7/a0-a6
          bra.s     CN.FIN
CN.nulo   moveq     #0,d0
          rts
CN.close  move.l    a0,-(a7)
          lea       $18(a0),a0
          lea       $140(a6),a1
          move.w    $D4,a2
          jsr       (a2)      desconectamos el canal de la lista encadenada
          move.l    (a7)+,a0
          move.w    $c2,a2
          jmp       (a2)      liberamos el bloque de canal del QDOS
CN.fin    suba.l     a0,a0  devolvemos 0 al QC
#endasm
}
desc_nul()
{
          base_nul;
#asm
          movem.l   d0-d7/a0-a6,-(a7)
          move.l    a0,-(a7)
          lea       $18(a0),a0
          moveq     #$23,d0
          trap      #1        desconectamos el controlador del QDOS
          move.l    (a7)+,a0
          move.w    $c2,a2
          jsr       (a2)      liberamos el bloque de control del QDOS
          movem.l   (a7)+,d0-d7/a0-a6
          suba.l     a0,a0     devolvemos 0 al QC
#endasm
}
```

Bueno... Ya tenemos aqui unas cuantas cositas más. Las rutinas de gestión de errores, con dos entradas unas que son avisos y otras que son catástrofes como vereis por una entrada el fichero objeto se cierra, se destruye y se crea un canal objeto al fichero dispositivo NUL dispositivo que se crea con la rutina de creación que aparece justito aquí encima.

También tenemos un programa principal provisiional, a la espera de que le enganche el QPTR y un programa COMPILA válido iniciallmente aunque probablemente se reestructure un poco dado que su diseño es bastante feo.

La identification división se puede quedar tal y como está dado que es válida y es estandard, a lo sumo se la pondría que escrbiera los comentarios en el ejecutable a fin de que se pudieran visualizar.

Una rutina muy importante es la de comprobación de palabras clave dado que la sintaxis del COBOL es muy estricta, una vez que se encuentra una palabra clave en un lugar del programa, se presupone que han de seguir otra serie de palabras justo a continuación ue aquí se comprueban.

Otra rutina importante es la que busca un punto de final de párrafo dado que el punto es el separador de estructuras COBOL.

Para que no os sintais perdidos mucho a continuación va la tabla de errores en inglés y en español con la que os podreis enterar de los multiples pufos que se van detectando.

```
  * ENGLISH MESSAGES VERSION  *
Please enter your command line?
    %d Errors      %d Warnings
      Object File not generated
******          PRESS ANY KEY TO STOP PROGRAM            ******
***ERROR 3  Unexpected end of file                            *
***ERROR 4  Close quotes not found                            *
***ERROR 5  Open quotes missing                               *
***WARNING 6  Can't find IDENTIFICATION DIVISION            @@@
***ERROR 7  Can't find PROGRAM-ID                             *
***WARNING 8  String after '.' not parsed                   @@@
***WARNING 9  Can't find ENVIRONMENT DIVISION               @@@
***ERROR 10 Bad IDENTIFICATION DIVISION syntax                *
***ERROR 11 ENVIRONMENT DIVISION missed;  parsing DATA DIVISION
***ERROR 12 DATA DIVISION missed; parsing PROCEDURE DIVISION  *
***WARNING 13 Can't find DATA DIVISION                      @@@
***ERROR 14 Parsing supossed ENVIRONMENT DIVISION             *
***ERROR 15 Margin «B» expected                               *
***ERROR 16 Bad ENVIRONMENT DIVISION syntax                   *
***ERROR 17 Margin «A» expected                               *
***ERROR 18 '.' expected                                      *
***ERROR 19 Parsing suposed IDENTIFICATION DIVISION           *
***ERROR 20 Parsing suposed DATA DIVISION                     *
***ERROR 21 Literal expected                                  *
***ERROR 22 Number expected                                   *
***ERROR 23 User-name expected                                *
***WARNING 24 User-name too long                            @@@
***ERROR 25 Not a user-name character                         *
***ERROR 26 "IS" Expected                                     *
***ERROR 27 undefined PROGRAM COLLATING SEQUENCE              *
***WARNING 28 Only the first literal character is estimated @@@
***ERROR 29 DIVISION expected                                 *


VERSION CON MENSAJES EN ESPAÑOL
Introduzca la línea de comando?
    %d Errores       %d Avisos
     Fichero objeto no generado
******         PULSE UNA TECLA PARA FINALIZAR            ******
***ERROR 3  Fin de fichero fuente inesperado                  *
***ERROR 4  Faltan comillas de cierre                         *
***ERROR 5  Faltan comillas de apertura                       *
***AVISO 6  No encuentro la IDENTIFICATION DIVISION         @@@
***ERROR 7  No encuentro el PROGRAM-ID                        *
***AVISO 8  Caracteres tras '.' no analizados               @@@
***AVISO 9  No encuentro la ENVIRONMENT DIVISION            @@@
***ERROR 10 Mala sintaxis en IDENTIFICATION DIVISION          *
***ERROR 11 ENVIRONMENT DIVISION extraviada; analizando DATA  *
***ERROR 12 DATA DIVISION extraviada; analizando PROCEDURE    *
```

```
***AVISO 13 No encuentro la DATA DIVISION                     @@@
***ERROR 14 Analizando supuesta ENVIRONMENT DIVISION            *
***ERROR 15 Esperado comienzo en margen «B»                     *
***ERROR 16 Mala sintaxis en ENVIRONMENT DIVISION               *
***ERROR 17 Esperado comienzo en margen «A»                     *
***ERROR 18 Esperado '.'                                        *
***ERROR 19 Analizando supuesta IDENTIFICATION DIVISION         *
***ERROR 20 Analizando supuesta DATA DIVISION                   *
***ERROR 21 Se esperaba un literal                              *
***ERROR 22 Se esperaba un numeral                              *
***ERROR 23 Se esperaba un nombre-del-usuario                   *
***AVISO 24 Nombre-de-usuario demasiado largo                 @@@
***ERROR 25 Caracter no permitido en nombre-del-usuario         *
***ERROR 26 Se esperaba un "IS"                                 *
***ERROR 27 PROGRAM COLLATING SEQUENCE no definida              *
***AVISO 28 Solo se valora el primer caracter del literal     @@@
***ERROR 29 Se esperaba una DIVISION                            *
```

Bien como os podeis imaginar por los mensajes que ya os muestro, en la proxima entrega, hablaremos de la ENVIRONMENT DIVISION otra división sencillita y sin grandes complicaciones al principio pero luego.... ya veremos.

Madrid 27 de Junio de 1989
NACHO ENRIQUE CABERO
C/MIGUEL DE LA IGLESIA 3 1ºc
28027 Madrid

COMENTARIOS DE PROGRAMAS

EDISC

PROGRAMA: Edisc
AUTOR: JComp (José Carlos de Prada)
LENGUAJE: SuperBasic compilado con QLiberator
PRECIO: gratuito para colaboradores y lectores del CUQ

Durante el mes de Julio, en plena época de exámenes tuve que echarle una ojeada a la organización interna de la información en los discos del sistema operativo Minix. Para poderlo hacer cómodamente en casa tuve que escribir unas breves rutinitas en Basic para el QL. Poco a poco fui ampliándolas, introduciendo nuevas funciones y mejorando el intercambio con el ususrio, a la vez que le daba una presentación más "comercial". El resultado final es el programa que ahora os presento. Mi deseo ha sido que llegase a los usuarios con el máximo grado de garantía en su funcionamiento, por lo que me he retrasado un mes desde su conclusión hasta enviároslo para poder depurarlo a conciencia. A pesar de ello algunos errores se me habrán escapado, teniendo en cuenta el tamaño del programa y el número de funciones con que cuenta. Espero que me informeis de cualquier fallo que encontreis para poder solucionarlo.

El núcleo principal de este artículo lo constituye el Manual de Instrucciones. Cualquier duda posterior puede ser aclarada a través del boletín o por carta a mi dirección.

Junto con este número del boletín del CUQ se incluye el programa Edisc totalmente listo para su uso. Es imposible presentar o comentar el código fuente en las páginas del boletín, dado que su longitud sobrepasa las 45 Ks.

EDISC
Manual de Instruccines para el Usuario

NOTA:
EDISC es un programa escrito por José Carlos de Prada.
El autor autoriza la distribución gratuita del programa y de sus

Se recomienda la lectura completa de estas instrucciones antes de utilizar EDISC con discos que contengan programas o información útiles para el usuario.

Deseo mostrar mi agradecimiento a Angel López, cuyas críticas acertadas han contribuido sin duda a mejorar el funcionamiento y la presentación de EDISC.

INDICE

1 - Presentación

EDISC es un editor de disco por sectores escrito especialmente para el QL y los controladores de disco de Tony Tebby.

El programa ha sido escrito en SuperBasic y compilado con QLiberator. En el proceso de compilación se han enlazado con el programa principal tanto QLib_run como QLib_ext, por lo que no es necesario cargar separadamente ninguno de ellos.

Para el desarrollo de EDISC se han utilizado distintas extensiones del SuperToolkit II de Care/QJump, siendo imprescindible que este último se encuentre residente en memoria en el momento de su ejecución.

Al diseñar EDISC se ha intentado obtener una herramienta útil y flexible para el trabajo con los discos en el QL. Con este fin se ha dotado al programa con un gran número de funciones que superan con creces las de otros programas anteriores y que le permiten responder a casi cualquier necesidad que se plantee en la manipulación de los discos.

En pocas palabras el espíritu de EDISC podría resumirse como "todo lo que siempre quisiste hacer con tus discos y nunca antes habías podido".

2 - Campo de Aplicación y Limitaciones.

EDISC corre en cualquier QL, incluso con 128 Ks, a condición lógicamente de que esté equipado al menos con una unidad de disco.

Como queda dicho, EDISC se ha programado para los controladores de disco de Tony Tebby (TrmpCard, SuperQBoard, QFLP, ...), no siendo adecuado para trabajar con controladores diferentes (MicroPeripherals original, ...).

Se ha tenido en cuenta la variedad de tipos de disco utilizados en el QL, por lo que EDISC es capaz de trabajar con discos de 40 y de 80 pistas indistintamente, así como con discos de 3"1/2 y de 5"1/4.

EDISC es también adecuado para los discos con formato MS DOS, siempre que cumplan las especificaciones anteriormente señaladas con respecto al número de pistas por cara.

Es posible trabajar con discos formateados en distintos sistemas, siempre y cuando se mantengan los equemas básicos en cuanto a número de pistas, número de sectores por pista y longitud de lo sectores. A parte de los discos formateados en QDOS y MS DOS se ha comprobado el correcto funcionamiento con los discos de Minix.

Siendo EDISC un programa compilado, puede ejecutarse indistintamente de forma independiente o en multitarea.

No es necesaria la utilización concurrente de ningún programa de los denominados manejadores de multitarea (multitaskers), aunque se obtendrá un funcionamiento más eficaz si se dispone del Window Manager (Wman) de QJump que se incluye en QRam y en QPointer. La diferencia básica cuando se ejecuta EDISC en multitarea sin este complemento consiste en la necesidad de redibujar la zona de pantalla del programa cada vez que se retorna desde otro programa que la haya alterado o borrado; para ello debe pulsarse MAYUSCULAS+INTRO desde el nivel de menú.

EDISC es compatible tanto con Lightning de Digital Precision como con SpeedScreen de Creative Code Works, obteniéndose importantes mejoras en la velocidad de ejecución con cualquiera de ellos.

3 - Estructura del Programa.

3.1 - Ventanas.

A lo largo del programa la información se muestra al usuario distribuida en seis ventanas con misiones específicas y diferentes.

```
+-----------------------------------------------------------------+
! 1   Edisc  v 1.0              © JComp   1989                    !
+-----------------------------------------------------------------+
!  Pos:    Dec:      ! 00 01 02 03 04 05  06 07 08 09 0A 0B 0D 0E 0F!
!00 2                ! 3                                           !
!01                  !                                             !
!02                  !                                             !
!03                  !                                             !
!04                  !                                             !
!05                  !                                             !
!06                  !                                             !
!07                  !                                             !
!08                  !                                             !
!09                  !                                             !
!0A                  !                                             !
```

```
!0B              !                                              !
!0C              !                                              !
!0D              !                                              !
!0E              !                                              !
!0F              !                                              !
+----------------+----------------------------------------------+
! 4  flp _         Cara:      Pista:     Sector:                !
+---------------------------------------------------------------+
! 5                                                             !
!                                                               !
!                                                               !
!                                                               !
+---------------------------------------------------------------+
! 6                                                             !
+---------------------------------------------------------------+
```

3.1.1 - Título y Copyright.

Se muestran de forma permanente durante la ejecución: el título del programa, el número de versión, el copyright y la fecha.

3.1.2 - Volcado ASCII.

Presenta cuatro zonas de información diferentes:

a) Zona Central. Ocupa la mayor parte de la ventana y muestra el volcado en código ASCII de 16 de las 32 líneas del sector actualmente seleccionado (cada línea contiene 16 bytes).

Cuando en la ventana de menús se presenta el menú S (Sector), se hace visible en esta ventana uno de los cursores no parpadeantes que indica la posición donde actuará cualquiera de los comandos seleccionados.

El volcado ASCII de la totalidad del sector se hace de forma continuada, o dicho de otro modo, si se lleva el cursor hacia abajo cuando se encuentra en la última línea de la ventana, las líneas actuales se desplazarán hacia arriba y aparecerá una nueva, pudiendo repetirse el proceso hasta alcanzar la última línea del sector (1F).

El mismo proceso pero en sentido inverso se lleva a cabo cuando se lleva hacia arriba el cursor situado en la primera línea.

Esta presentación continuada difiere de la utilizada en programas anteriores, permitiendo una visión de conjunto de la zona central del sector.

b) Zona Izquierda. En la franja izquierda se muestran los números de línea de las actualmente en pantalla.

La numeración se hace en hexadecimal y su rango va de 00 a 1F (de 0 a 31 en decimal).

c) Zona Superior-Izquierda. Se muestra la posición del cursor con respecto al inicio del sector. La numeración se hace en decimal y su rango va de 0 a 511.

d) Zona Superior-Derecha. Se muestra el valor decimal del byte actualmente seleccionado por el cursor.

3.1.3 - Volcado Hexadecimal.

Presenta dos zonas de información:

a) Zona Central. Ocupa la mayor parte de la ventana y muestra el volcado hexadecimal de 16 de las 32 líneas del sector actualmente seleccionado.

Cuando en la ventana de menús aparece el menú S (Sector), se hace visible en esta ventana el segundo de los cursores no parpadeantes que indica la posición donde actuará cualquiera de los comandos seleccionados.

Los cursores de las ventanas 2 y 3 se mueven al unísono, señalando posiciones equivalentes.

b) Zona Superior. Se muestra el número de las 16 columnas de cada fila. La numeración se hace en hexadecimal, siendo su rango de 00 a 0F (de 0 a 15 en decimal).

3.1.4 - Parámetrs Basicos.

Se muestran los cuatro parámetros básicos sobre los que se realiza el trabajo:

===============================

Parámetro	Rango
flp	1 - 2
Cara	0 - 1
Pista o	0 - 39 0 - 79
Sector	0 - 8

(Se ha elegido la numeración de 0 a 8 para el sector, en lugar de 1 a 9 por coherencia con el resto de parámetros).
Esta ventana no sólo muestra los datos, sino que también se utiliza para su entrada.

3.1.5 - Menús.
Muestra el menú de comandos vigente.
En la esquina superior izquierda, debajo del cursor parpadeante, aparece la letra que denomina cada menú. El resto de la ventana se utiliza para indicar los comandos disponibles y las teclas adecuadas para seleccionar cada uno de ellos.
Los comandos de la primera,tercera y cuarta líneas se obtienen mediante la/s tecla/s indicada/s a su izquierda en blanco. Los comandos de la segunda línea se obtienen pulsando la tecla de mayúsculas (o SHIFT) más la/s tecla/s indicada/s a su izquierda en blanco.
Existen dos menús diferentes cuyos comandos se describen detalladamente en las secciones 4.1 y 4.2.

3.1.6 - Entradas y Mensajes.
Se utiliza en general para mostrar los mensajes de error y para solicitar las entradas de datos en los comandos que lo requieren.

3.2 - Funcionamiento General.
EDISC se arranca en la forma ordinaria para cualquier programa ejecutable mediante el comando

EXEC FLP1_EDISC
o
EX FLP1_EDISC

Una vez arrancado, el programa abre todas sus ventanas manteniendo en blanco las número 5 y 6, así como las zonas centrales de las número 2 y 3, al no existir ningún sector seleccionado, ni tampoco ningun menú vigente.
En la ventana de parámetros básicos (4) aparecen los valores por defecto:

flp1_ Cara: 0 Pista: 00 Sector: 0

Sobre el valor del número de unidad aparece un cursor parpadeante. Para seleccionar un nuevo valor basta con teclear directamente un número del 1 al 2 o simplemente pulsar INTRO para seleccionar el valor por defecto.
Cuando se selecciona un valor el cursor pasa al parámetro siguiente a la vez que el valor seleccionado aparece en su lugar en color blanco.
Al seleccionar el número de la unidad de disco se podrá observar como ésta se pone en marcha. Es normal que se aprecie un lapso de tiempo más largo si el disco encontrado es de 40 pistas por cara y no de 80 como es habitual. Este retraso se debe al hecho de que EDISC comprueba este dato buscando un sector en la pista número 41 de la cara 1; si no lo encuentra el sistema operativo repite en varias ocasiones la búsqueda hasta convencerse de que no existe.
Podría haberse seguido un método más rápido para obtener este dato leyéndolo directamente del sector 0, pista 0, cara 0, donde lo almacena el sistema operativo, pero ésto habría supuesto limitar el uso de EDISC a los discos con formato QDOS.
Un retraso similar y por razones semejantes puede producirse si la unidad de discos está vacía cuando se selecciona.
Los restantes tres valores se seleccionan de forma parecida, teniendo en cuenta que en el caso de la pista, si el valor es menor que 10, será necesario seleccionar un 0 en primer lugar pulsando la tecla 0 o la tecla INTRO.
Cuando los cuatro parámetros básicos han sido seleccionados EDISC lee el correspondiente sector del disco, visualiza en las ventanas 2 y 3 lo volcados en ASCII y hexadecimal de sus primeras 16 líneas y se hace vigente el menú S, mostrándolo en la ventana 5, a la vez que aparecen en pantalla los cursores de posición ya mencionados.

La posición inicial dentro del sector es la 0.
A partir de este momento están disponibles todos los comandos del menú S,incluido el cambio de menú.
Una vez finalizado el trabajo, puede darse por terminada la ejecución de EDISC medinte el comando Fin del menú D (ver sección 4.2.5).

4 - Comandos.
EDISC dispone de 29 comandos o funciones diferentes. Su presentación al usuario se hace mediante dos menús; en cada uno de ellos se muestran en blanco la tecla o teclas que deben pulsarse para su obtención y el nombre del comando en negro.
Un comando sólo está disponible si se encuentra en el menú visible en la ventana 5.
En general todos los comandos que requieren de una entrada de datos para su ejecución la solicitan por la ventana 6 y pueden interrumpirse por medio de la tecla ESC antes de ser lanzados por la terminación de la entrada de datos.
En los comandos que requieren de una entrada de datos no es posible el retroceso del cursor para hacer correcciones, por lo que, si se ha cometido algún error es necesario pulsar ESC y volver a comenzar con el comando.

4.1 - Menú S.
El menú S (Sector) agrupa todos los comandos que actúan dentro del ámbito del sector actualmente seleccionado, más un grupo de comandos que sirven para la "navegación" entre sectores y entre pistas. De esta manera pueden llevarse a cabo desde este menú todas las tareas de edición dentro de un sector o dentro de un fichero, aunque EDISC no dispone de funciones para el manejo directo de ficheros (para la organización de los ficheros en el disco consúltese la bibliografía de la Nota Final).

Comandos del Menú S:

4.1.1 - Menu D. (F1) Permite el paso al Menú D.

4.1.2 - Posición 0. (F2) Sitúa los cursores de las ventanas 2 y 3 en la posición 0, visualizando la primera mitad del sector actual.

4.1.3 - Posición 511. (F3) Sitúa los cursores de las ventanas 2 y 3 en la posición 511, visualizando la segunda mitad del sector actual.

4.1.4 - Sector +. (F4) Avanza al siguiente sector. Los cursores de las ventanas 2 y 3 quedan situados en la posición 0.

4.1.5 - Sector -. (F5) Retrocede al sector anterior. Los cursores de las ventanas 2 y 3 quedan situados en la posición 0.

4.1.6 - Buscar. (MAYUSCULAS+F1) Busca una cadena de caracteres a lo largo del sector actual, comenzando desde la posición 0. La entrada de la cadena de caracteres puede hacerse en caracteres ASCII o en hexadecimal. En este último caso la cadena debe ir precedida por el símbolo $ y cada par de dígitos hexadecimales se interpreta como el código correspondiente a un caracter. Si el número de caracteres introducidos es impar se supone la existencia de un 0 como primer dígito. Si la entrada de la cadena se hace en caracteres ASCII y el primero ha de ser el caracter $, debe precederse de un apóstrofe (') para evitar que sea interpretado como el principio de un código hexadecimal. El caracter apóstrofe (') al principio de cadena hace que se interprete textualmente la cadena que le sigue, de forma que si el primer caracter ha de ser un apóstrofe deberán incluirse dos apóstrofes seguidos en lugar de uno; de lo contrario, éste sería desechado por el programa antes de iniciar la búsqueda.
Si se contesta con la tecla INTRO a la petición de entrada de cadena, EDISC continúa buscando la cadena de la búsqueda anterior a partir del punto en que se encontró; si se ha cambiado de sector desde la última búsqueda, se continúa a partir de la posición 0.

4.1.7 - Posición ? (MAYUSCULAS+F2) Sitúa los cursores en la posición seleccionada dentro del sector actual. Cualquier posición requiere tres cifras para ser seleccionada, por lo que valores entre 99 y 10 deberán llevar un 0 delante y valores inferiores a 10 deberán llevar dos 0s delante. La posición

por defecto es la 0 (000), pudiendo seleccionarse mediante pulsaciones consecutivas de la tecla "0" o de la tecla INTRO indistintamente. Los valores superiores a 511 son rechazados automáticamente.

4.1.8 - Cara. (MAYUSCULAS+F3) Permite seleccionar el sector de idéntico número y pista que el actual pero en la cara opuesta.

4.1.9 - Sector 3 + (MAYUSCULAS+F4) Avanza tres sectores. Este comando, así como Sector 3 -, Pista 6 + y Pista 6 -, junto con Cara, permiten la "navegación" a través de los sectores de un mismo fichero, según la distribución que hace el sistema QDOS.

4.1.10 - Sector 3 - (MAYUSCULAS+F5) Retrocede tres sectores.

4.1.11 - Cursor. (TECLAS DE CURSOR) El movimiento del cursor se obtiene mediante la pulsación de las cuatro teclas de cursor:

Arriba, Abajo - permiten el movimiento entre filas a lo largo de una columna. Un intento de llevar el cursor más allá de la última fila del sector lo coloca en la última posición. Un intento de llevar el cursor más arriba de la primera fila lo coloca en la primera posición.

Derecha, Izquierda - permiten el movimiento a lo largo de una fila. Tras superar la última posición de una fila, el cursor reaparece al principio de la siguiente.

4.1.12 - Editar (INTRO) Permite modificar el valor de uno o varios bytes del sector actual. La introducción del nuevo valor se hace a través de la ventana 6. Cada valor puede introducirse como caracter ASCII o como número hexadecimal, decimal o binario. El primer caracter es el que determina la interpretación que se haga del valor, según el código siguiente:

$ - hexadecimal
- decimal
% - binario
' - interpretación literal de la cadena siguiente

Cualquier caracter que no vaya precedido de ninguno de los anteriores se interpreta como ASCII. Para introducir como ASCII alguno de los símbolos especiales debe anteponerse el apóstrofe: '$ '# '% ''.

Si se introduce una cadena de caracteres más larga que la necesaria para determinar el valor de un sólo byte, se toma como válida solamente la primera parte de la cadena.

Una vez introducido el nuevo valor se pulsa INTRO, con lo que se sitúa en el byte señalado por los cursores de posición; los cursores avanzan una posición y y el programa queda a la espera de recibir un nuevo valor para otra posición.

Es posible mover los cursores de posición con las teclas de cursor mientras se encuentra seleccionado el comando Editar.

Para salir del comando Editar se utiliza la tecla ESC.

Cualquier modificación producida en un sector se realiza directamente sobre el buffer en memoria RAM que contiene una copia del sector y no sobre el sector mismo en el disco. Por lo tanto la modificación definitiva se produce al abandonar el sector para pasar a otro o para abandonar el programa, por lo que antes de producir dicha operación se pide conformidad en la ventana 6.

4.1.13 - Refresco. (MAYUSCULAS+INTRO) Permite "redibujar" la pantalla cuando ésta ha sido modificada por otro programa que se ejecute en multitarea y no se dipone del Window Manager de QJump.

Es el único comando común a los dos menús.

4.1.14 - Entero. (ESPACIO) Devuelve en la ventana 6 los valores de los dos bytes situados a la izquierda y bajo los cursores de posición, considerados como entero de 16 bits sin signo y como entero de 16 bits con signo. También se obtiene el valor de los tres bytes situados a la izquierda y bajo los cursores de posición, considerados como un entero de 24 bits sin signo.

4.1.15 - Pista + (P) Se avanza al sector de idéntico número, situado en la pista siguiente de la misma cara.

4.1.16 - Pista - (CTRL+P) Se retrocede al sector de idéntico número, situado en la pista anterior de la misma cara.

4.1.17 - Pista 6+ (MAYUSCULAS+P) Se avanza al sector de idéntico número, situado seis pistas más adelante en la misma cara.

4.1.18 - Pista 6- (MAYUSCULAS+CTRL+P) Se retrocede al sector de idéntico número, situado seis pistas más atrás en la misma cara.

4.2 - Menú D.

En el menú D (Disco) se reunen todos aquellos comandos que actúan a nivel de Disco, a excepción de los que sirven para la "navegación" entre sectores o pistas. Además se encuentra repetido el comando de refresco de la pantalla y existe un comando para control de los avisos sonoros.

Comandos del Menú D:

4.2.1 - Menú S. (F1) Permite el paso al Menú S (Sector).

4.2.2 - Nuevo Sector. (F2) Selecciona un nuevo sector por medio de la ventana de Parámetros Básicos (4). El proceso de selección de dichos parámetros se lleva a cabo de idéntico modo al llevado a cabo al inicio del programa (ver sección 3.2).

4.2.3 - Crear. (F3) Permite crear y abrir un fichero en un dispositivo (MDV_ o FLP_) diferente del que contiene el disco sobre el que se está trabajando. La finalidad principal de dicho fichero es la de copiar en él sectores aislados seleccionados por el usuario, pudiéndose de esta manera recuperar ficheros borrados o deteriorados en parte. En todo caso este proceso no se realiza por parte del programa de una forma automática, sino de forma totalmente manual por parte del usuario, quedando bajo su responsabilidad el resultado final, aunque con un mayor nivel de flexibilidad en la operación.

4.2.4 - Cerrar. (F4) Cierra el fichero abierto con el comando Crear.

4.2.5 - Fin. (F5) Abandona el programa definitivamente. Si existe un fichero auxiliar abierto, se cierra automáticamente. Si se han producido modificaciones en el sector actual, se pide confirmación para la grabación de dichas modificaciones en el disco.

4.2.6 - Buscar. (MAYUSCULAS+F1) Busca una cadena a lo largo del disco. La introducción de la cadena a buscar se realiza por la ventana 6 y de forma idéntica a la de la búsqueda en el sector. El punto de partida de la búsqueda puede seleccionarse por parte del usuario, siendo el sector de inicio por defecto el 0 (sector 0, pista 0, cara 0). La búsqueda continúa hasta que se detecta la presencia de la cadena, se alcanza el final del disco o se pulsa la tecla ESC.

Si se pulsa la tecla INTRO en lugar de introducir una cadena, la búsqueda see produce sobre la cadena introducida en el último comando Buscar del Menú D y a partir del punto en que éste se detuvo.

Cuando se localiza con éxito la cadena buscada, se selecciona automáticamente el Menú S.

4.2.7 - Nuevo Disco. (MAYUSCULAS+F2) Cierra el disco de trabajo actual y solicita datos en la ventana 4 para abrir uno nuevo.

4.2.8 - Copiar. (MAYUSCULAS+F3) Copia un sector en el fichero auxiliar abierto con el comando Crear. Los datos del sector a copiar se solicitan en la ventana 6, siendo los parámetros por defecto los del sector actual.

4.2.9 - Sector Absoluto. (MAYUSCULAS+F4) Permite seleccionar un nuevo sector, introduciendo el número que ocupa en la cuenta total en el disco, considerando como sector 0, el sector 0, de la pista 0, de la cara 0. El máximo sector disponible es 1439 o 719, según el caso.

4.2.10 - Posición Absoluta. (MAYUSCULAS+F5) Permite seleccionar una posición en cualquier punto del disco, introduciendo por la ventana 6 el número que dicha posición ocupa en la cuenta total del disco, considerando como

posición 0, la posición 0, del sector 0, de la pista 0 de la cara 0. El máximo valor admitido es 737279 o 368639, según el caso.
Queda seleccionado automáticamente el Menú S.

4.2.11 - Refresco. (MAYUSCULAS+INTRO) Ver sección 4.1.13.

4.2.12 - Sonido. (S) Habilita o inhabilita el aviso sonoro en caso de error o búsqueda fallida. La situación inicial es de sonido habilitado.

5 - Nota Final.

Con el uso de EDISC es posible manipular de forma efectiva los discos. Antes de proceder a modificar su contenido es imprescindible tener un conocimiento mínimo de la organización de la información; el propio EDISC puede ser un buen camino para lograr dicho conocimiento. Para una información más concreta puede utilizarse el artículo "Como en Botica" de Ignacio Enrique Cabero en QLave no 3 Vol. IV de Septiembre de 1.987, en lo referente al sistema operativo QDOS.

En cuanto al Sistema Operativo Minix puede consultarse el capítulo 3.6 del libro de Andrew S. Tanenbaum "Sistemas Operativos. Diseño e implementación", publicado en castellano por Prentice-Hall, México, 1.988.

Respecto al sistema MS-DOS puede encontrarse alguna referencia en el mismo libro de Tanenbaum y en otros de entre la abundante bibliografía existente en el mercado.

Para cualquier duda o si se detecta algún fallo en el correcto funcionamiento de EDISC puede referirse a:

José Carlos de Prada
C/ Hacienda No 16
28019 Madrid

Z88

Z88 USERS' CLUB

Desde septiembre de 1987 existe en UK el Z88 USERS' CLUB, que edita su revista Z88 EPROM bimensual de 34+ páginas. Una subscripción a 6 números para España cuesta 9 libras (¡Bastante económico!). Su dirección es:

Z88 USERS' CLUB
69 WELLINTGTON STREET
LONG EATON
NOTTINGHAM
NG10 4NG

El editor/presidente se llama Roy A. Woodward, y debe ser un loco coleccionista Sinclairiano, porque en todos los últimos Z88 escribe en la sección pequeños anuncios: **WANTED** For private colletion - any old Sinclair products up to ZX80.

La mayoria de los socios son usuarios de Sinclair QL. Y no es de extrañar que la libreria de programas para Z88 fuese suministrada en un principio solamente en EPROMs y MDV/FLP QL. Desde Mayo 1989, también es posible en discos en formato PC y ST, o en cinta de Cassette usando el Z-Tape (es muy posible que el librero siga usando el QL por su facilidad a la hora de usar discos PCs y STs de 3.5" o 5.25"). De todas formas la libreria de software no ha tenido un crecimiento importante de titulos hasta hace muy poco (tanto que muchos socios se han hecho la pregunta de si el Z88 iba a tener el mismo futuro que el viejo QL) siendo el número de titulos actualmente de 81 y aumentando muy rapidamente. Entre los titulos podemos encontrar juegos gráficos,

un paquete de estadistica (con 41 programas sumando un total de 70K), desensamblador Z80, Comunicación modem tipo Hayes, Control stock, Analisis financiero, Decisión de mercado, Z88fax, etc....

La revista Z88 EPROM ha sido siempre confeccionada e impresa usando un Z88 y una impresora Láser, pero desde el número 12, Roy ha decidido usar su Amstrad PC1640 transferiendo todo el texto con PC LINK II, y usando ese programa para traducir el formato Pipedream a Wordstar. Luego el fichero Wordstar es entonces importado a un desktop publishing (Timeworks DTP) donde se hace el enmaquetado final. Según Roy, este nuevo sistema es muchisimo más laborioso y consume muchisimo más tiempo que el anterior sistema, pero dice que vale la pena (yo no veo casi ninguna diferencia a excepción de que parece que se ha conseguido aumentar el número de lineas por página).

La base de datos de socios y la impresión de las etiquetas es gestionado por un viejo QL y una vieja impresora matricial.

El contenido de la revista Z88 EPROM es casi el mismo que en nuestro CUQ. Tiene su editorial, Helpline, cartas abiertas, noticias, comentarios de software, hardware y libros, Basic para principiantes, compra-venta-contactos,...... Pero con la diferencia de que en el último número los anuncios pagados ocupan casi la mitad de la revista (¡Esto ya parece Sinclair QL World!).

Existe otro CLUB Z88 USA que últimamente está haciendo engordar bastante la libreria de programas de Z88 USERS' CLUB. Pero es muchisimo más caro, 6 números cuestan 32.95 dólares (CLUB Z88, Time Designs, 29722 Hult Road, Colton, Oregan 97017, U.S.A.).

S. Merino, 25/7/1989.

UN PAR DE SISTEMAS DE DISCOS PARA EL Z88

Todo este articulo está basado en la información publicada en Z88 EPROM, la cual es muy completa, y el motivo es el mismo de siempre, no tengo acceso a ninguno de los dos sistemas de discos comentados.

Por lo visto el número de usuarios de Z88 que no poseen otro ordenador está aumentando y los cartuchos Eprom, aunque sean muy seguros y etc, son muy caros (1K cuesta 40p). Para aquellos que poseen otro ordenador personal, no hay problema, pueden transferir los ficheros al otro ordenador, pero cuando no tenemos otro ordenador la cosa se complica.

Una solución podria ser el Z-Tape, que es una combinación de Hardware y Software que permite almacenar los ficheros en cintas de cassette normales. Pero ese sistema además de lento, tiene todos los problemas de poca seguridad de todos los sistemas de almacenamiento en cinta.

Lo que el Z88 siempre ha pedido a gritos es un sistema de almacenamiento en disco. La cuestión es cuanta gente no tiene acceso a otra máquina, y cuantos están dispuestos a pagar más de lo que cuesta un Z88 por sistema de almacenamiento de disco.

Actualmente se puede encontrar en el mercado dos sistemas de discos basados prácticamente en la misma tecnologia, PCIC Portable Disc Drive y XOB Disc-88 Portable Disc Drive.

Ambos sistemas están basados alrededor de un disc drive que ya existia, el cual es vendido por Tandy (hecho en Japón por Brother) y se usa con sus ordenadores portatiles modelos 102 y 200. El interface de disco de esas máquinas es RS232 serial port, y aqui es principalmente donde comienza la primera compatibilidad con el Z88.

Ambos drives son suministrados en el embalaje original de Tandy. Conteniendo manual de instrucciones, Tandy disc (conteniendo el DOS para ordenadores Tandy) y un cable. El color del drive es crema (seguramente el color de los portatiles Tandy). El modelo de PCIC ha sido modificado interiormente, pero el de XOB sigue siendo el Standard Tandy. El disco puede funcionar con 4 pilas iguales a las usadas por el Z88 o con un adapatador de corriente idéntico al del Z88, pero con la diferencia de que la polaridad es opuesta.

El drive usa discos 3.5" de simple cara, dando un formateo de 202.24Kbytes. La máxima longitud de un fichero está limitada por hardware a 64K.

El Software controlador del disc drive es suministrado en un cartucho Eprom de 32K, el cual es leido desde el BASIC y podemos retirarlo una vez leido, pudiendo tener el slot libre para otra cosa.

El programa PCIC es llamado ZN-DOS, y tiene una longitud de 2769

bytes, mientras que el XOB Disc Manager tiene 5632 bytes.
El menu del ZN-DOS es:
Bank 0 Directory Format Kill Load Quit Rename Save View
El menu del Disc Manager es:
CAT DELETE SAVE LOAD RENAME FORMAT BANK=0 QUIT
El disco tiene una capacidad de 200K, y puede ser dividido en 2 discos virtuales, Bank 0 y Bank 1. Cada banco solamente permite un máximo de 40 ficheros, y cada banco tiene su propio directorio.
El tiempo de ejecución usando un fichero de 5377 bytes es:
FORMAT: 110 secs CATALOGUE: 2 secs SAVE: 25 secs LOAD: 14 secs (PCIC) o 7 secs (XOB) DELETE: 7 secs RENAME: 2 secs
Los tiempos son idénticos para ambos sistemas excepto en LOAD.
Hay que aclarar que esto es un sistema backup en disco. Los ficheros se guardan del FILER al disco y se leen del disco al FILER (el Z88 está especializado en trabajar con Ramdisk).
Ambos sistemas son compatibles en formato de disco. Pero al haber sido alterada la disquetera de PCIC, no pueden intercambiarse la disquetera o software.
PRODUCTO: PCIC Portable Disc Drive
PRECIO : 275 libras incluido VAT y Transporte
DISTRIBUIDOR: TK Computerware, Stone Street, North Stanford,
Ashford, Kent. TN25 6DF

PRODUCTO: Disc-88 Portable Disc Drive
PRECIO : 255 libras incluido VAT y Transporte
(si alguien posee ya un Tandy Portable Drive 2, XOB le puede vender el cable adaptador y el software para el Z88 por 64.50 libras incluido VAT y Transporte).
DISTRIBUIDOR: XOB, Balkeerie, Eassie, By Forfar, Angus. DD8 1SR

Según mi opinión, conociendo el protocolo de comunicación, ese disco portatil podria ser usado en un QL también, escribiendo el software.
S. Merino, 27/7/1989

ESPERANTO, LA LENGUA INTERNACIONAL (Capítulo III)

En espera de ultimar algún material en relación tanto al Esperanto como a la informática (combinación muy en boga en la actualidad en las investigaciones en punta en los terrenos de la traducción por ordenador y la comprensión del lenguaje natural), en este capítulo de la serie incluyo la reproducción de un interesante artículo sobre el Esperanto a nivel general:

DOCTOR, SOY UN POBRE DIABLO...

-Compréndame, doctor, ¡son esos altibajos los que me matan!

-¿Cuándo aparecieron los primeros síntomas?

-No hace ni cien años que estuve durante tantos milenios, ¡tan tranquilo! Sólo una ínfima parte podía comunicarse por encima de las barreras lingüísticas. Mi patrono Lucifer estaba contento: los hombres no se entendían de uno a otro pueblo.

-¿Qué sucedió en el siglo pasado?

-Yo tuve mucho miedo. ¡Por culpa de un chiquillo! Ello sucedió en Bialistok, en lo que hoy es Polonia. El miraba con sus grandes ojos doloridos a las gentes de su ciudad que se destrozaban: cuatro etnias (polacos, judíos, rusos, alemanes), cuatro alfabetos, cuatro religiones, cuatro odios. El hablaba corrientemente las cuatro lenguas locales y sabía por experiencia que, franqueado el puente lingüístico, las relaciones se establecerían sobre sobre una base distinta. Tenía todo lo necesario para desarticular mi plan: era un superdotado, como se dice hoy, que tenía corazón y una sensibilidad artística muy fina. Para colmo, era a la vez modesto y tenaz. ¡La peor de las combinaciones humanas! Vi dibujarse su resolución en la cara. Estudié los cuadernos donde consignaba su proyecto. En lingüística, llevaba un siglo de adelanto sobre su época. Fui presa del pánico.

-¿Qué había de extraordinario en sus ideas?

-Había comprendido él que, para el diálogo entre extranjeros, era preferible utilizar una lengua nacida a nivel internacional, que la lengua de un pueblo

dado. Pero, sobre todo, había comprendido que una lengua surge necesariamente de un proceso social, colectivo y anónimo. Miles de proyectos de "lengua internacional" o de "lengua universal" habían sido ya publicados, por ejemplo, por Descartes y Leibniz, pero sus autores no habían sentido lo que era una lengua y habían propuesto unos códigos rígidos, prefabricados, por decirlo de algún modo. Luis Zamenhof, a los 17 años, sabía que una lengua se constituye en gran parte inconscientemente, por el uso. Para él, una lengua era un organismo vivo, que no cesa de captar elementos en el medio exterior, asimilarlos, convertirlos en materiales para su propio crecimiento y desprenderse de los que no se integren a su genio particular. Así pues, se ciñó a organizar un pequeño núcleo de elementos lingüísticos y a proponer algunas estructuras fundamentales concebidas para que una lengua viva pueda nacer de manera neutral, si por ventura un grupo de gentes lo adoptase para comunicarse.

-Pero, ¡no había ninguna posibilidad de que se adoptase un código tan rudimentario!

-En efecto; es por lo que le hablo de altibajos. Cuando vi que los editores rechazaban publicar su proyecto (unas centenas de raíces, 16 reglas gramaticales, unos ejemplos de textos), me dije: "Falsa alarma". Por desgracia, nuestro hombrecillo consiguió editar su ridículo folleto a cuenta de autor. No era notable, pero en mi calidad de diablo especializado en cuestiones lingüísticas, sé oler al genio, y todas las ideas del chiquillo eran geniales. Tuve mucho miedo.

-¿Qué hizo usted?

-Nada, por de pronto. Conté con la inercia humana. La pega estaba en que la Europa del Este, al final del siglo XIX, era un lugar de gran efervescencia social y cultural nadando en un clima de idealismo apasionado. Con el inicio de las ideas sindicalistas, nuevas manifestaciones solidarias aparecían en toda una capa de la población, entre las más modestas; se manifestaban unas curiosidades culturales que jamás habían existido antes. Por otra parte, esas gentes trabajaban duro y no disponían de tiempo para ponerse a estudiar el francés o el inglés.

-¿Hubo, pues, cierta coincidencia entre la propuesta de Zamenhof y la necesidad de comunicación de cierto número de individuos?

-Exactamente. Bastaba con recorrer el folleto para darse cuenta de que después de unas horas de estudio se podía comenzar a tener correspondencia. ¡Imagínese lo que supone para un almacenista de los talleres del ferrocarril de Riga, poder tener correspondencia directamente con sus colegas de Estocolmo y de Budapest sin tener que aprender una lengua complicada! ¡Era una extraordinaria apertura hacia el mundo! La noticia se extendió como un reguero de pólvora. Lo que temía se produjo: unas gentes adoptaron el pequeño núcleo lingüístico para comunicarse entre ellos.

-Pero, ¡usted no permanecería sin reaccionar!

-No. Yo les ridiculicé. El envite del asunto era de carácter social, y me eché al bolsillo a los intelectuales. Sus privilegios se apoyan en cierta medida en el conocimiento de las lenguas y en una especie de monopolio sobre la comunicación internacional. Ellos me apoyaron admirablemente, con una mala fe que me deleitó.

-Y gracias a eso, el proyecto de Zamenhof nació muerto, como todos saben.

-Pues no; precisamente no murió. En 1902, cuando apareció la primera lista de personas que lo utilizaban con regularidad, se pudo encontrar nombres de chinos, japoneses, mongoles, sudamericanos, escandinavos, eslavos e incluso, en Tananarive, un cierto señor Ravelojaona. El mal había prendido por todas partes.

-Esas gentes no eran, sin embargo, sino una ínfima minoría, increiblemente diseminada.

-Desde luego. Pero los lazos que les unían eran de una fuerza rara. Ellos habían aprendido el embrión lingüístico para comunicarse y se comunicaban. Se visitaban unos a los otros, cotizaban para enviar un delegado a una reunión internacional en un país vecino, organizaban campos, encuentros, reuniones de toda especie, publicaban juntos boletines, revistas. En 1913, dos revistas en la lengua internacional lanzada por Zamenhof aparecían en los círculos obreros de China, ¡dése usted cuenta!

-Pero entonces, la lengua...

-Si; la idea de Zamenhof se convertía en realidad. Cuando una palabra o una expresión era necesaria, alguien proponía una u otra forma. Si tenía éxito, entraba en la lengua. Si no, se la desechaba. Eso se hizo tanto para el vocabulario como para la gramática. Lo que hoy se llama Esperanto es el resultado de esta práctica anónima. Por ejemplo, una "cartera" en el sentido de "portadocumentos", se dice TEKO, pero nadie sabe dónde, cuándo, en qué país, por quién, fue introducida esta palabra. Puede leer las obras completas de Zamenhof -quien, consciente del papel importante de los escritores en la formación de una

lengua, publicó miles y miles de páginas- y no la encontrará.

"Pero, aún se produjeron cosas mucho peores. Un día, vi a un joven polaco enamorarse de una muchacha italiana en un campo de esperantófonos. Su sola lengua común era el Esperanto. El modo por el cual este muchacho explotaba todos los recursos de la lengua para expresar su amor era inimaginable. Un verdadero poeta. Cuando se casaron y tuvieron hijos cuya lengua materna era el Esperanto, yo estaba desquiciado, atolondrado. Si el público, por azar, se diera cuenta de lo que es esta lengua VIVA, que no está ligada a ninguna etnia, a ninguna ideología, a ningún país, y que se aprende en diez veces menos tiempo que una lengua nacional, exigiría que se la enseñase por todas partes en la escuela primaria.

-¿Por qué en la escuela primaria?

-Es la tesis de los partidarios del Esperanto. Pedagógicamente, se ha comprobado que el niño aprende esta lengua extremadamente deprisa, al ser totalmente regulares las estructuras gramaticales y léxicas, y que ello le hace ganar un año o dos en el aprendizaje ulterior de otras lenguas. En el nivel secundario, el niño aprendería una lengua de su elección -inglés, español, persa, chino, latín...- por interés cultural, puesto que el papel de instrumento internacional de comunicación estaría reservado al Esperanto.

-Los niños aprenden muy deprisa no importa qué lengua. ¿No teme nada?

-¡Ay!, doctor, usted se equivoca. Se nota que usted no lee las reseñas de los expertos. Se efectuaron unas experiencias en ciertas escuelas primarias de Suiza: al cabo de tres años los niños no tenían más que una escasa comunicación y los observadores notaron un desánimo desde el primer año, cuando los niños se dieron cuenta de la lentitud de sus progresos. En Irán, el ministro de educación nacional se quejó el año pasado de que después de seis años de inglés, la mayoría de los alumnos eran incapaces de escribir una frase sin tener faltas.

-¡Pero en Esperanto, se encuentra la misma dificultad!

-Precisamente, no, y es lo que me exaspera. Usted no imagina lo que es una lengua cuyas estructuras son totalmente coherentes, donde basta saber "pan", PANO, para poder formar por sí mismo "panadero", PANISTO, y "panadería", PANEJO, donde basta saber "sano", SANA, para formar por sí mismo "curar", RESANIGI, y "enfermo", MALSANULO, una lengua cuya gramática se asimila en unas horas...

-Debe ser una lengua muy pobre.

-¡Ah, doctor, si pudiera decir verdad! Es lo que hago decir a los caracteres fuertes, pero de vez en cuando hay personas de buena fe que quieren juzgar por sí mismas. Ellos descubren que la limitación de medios no quita nada a la riqueza, al contrario. En español, las praderas verdean y los campos de trigo amarillean, pero los campos de berenjenas no pueden "moradear" o "rosear" un rosal. Veamos. He aquí una muestra, comprensible después de sólo unas lecciones, pero que no tiene equivalente en las numerosas palabras en español: HO PLEJ KISINDA! PLEJ AMINDA! NUR VIDETI VIN SAGAS MIN REKTE AL LA KOR'! (EJ se pronuncia como "ey" en "ley"). Es intraducible. Eso quiere decir algo así como "Oh tú, quien más que cualquier otra mereces ser abrazada, besada, tú, quien más que cualquier otra eres digna de ser amada, el solo hecho de vislumbrarte me lanza una flecha recta al corazón". Si leyera a los poetas y novelistas que escriben directamente en Esperanto, vería hasta qué punto esta lengua se ha hecho expresiva al mismo tiempo que conserva su sencillez.

-Pero, ¿es exacta?

-¡Ay, sí! Cuando se dice en inglés "Japanese encephalitis vaccine", el profano no sabe si se trata de un bacilo fabricado en Japón contra la encefalitis, o de un bacilo específico de la enfermedad llamada "encefalitis japonesa". En Esperanto se dice JAPANA ENCEFALITVAKCINO en el primer caso, JAPANENCEFALITA VAKCINO en el segundo; eso no tiene la pesadez del francés o español, obligados a agregar preposiciones, aunque carezca de la fluidez del inglés, en el que nunca se sabe a qué se refieren las palabras.

-De cualquier modo, no tiene por qué preocuparse, puesto que el Esperanto nunca fue tomado en serio por nadie.

-No es tan fácil, doctor, y eso eso es lo que me hace sufrir. Siempre los altibajos. Faltó poco para que fuera adoptado en la Sociedad de Naciones en 1922. El informe del Secretariado fue totalmente favorable. Felizmente, los chauvinismos de las grandes potencias pusieron orden en ello. Pero, cuántos chantajes, cuántas presiones no tuvieron que ejercer sobre los paises pequeños.

-¡Todo eso ya es viejo!

-Pero sufrí muy fuertes conmociones. Fue un "bajo". Por suerte, poco tiempo después conocí un "alto": los esperantófonos fueron en gran parte exterminados, al ser su lengua asimilada, aquí, a una maniobra del "cosmopolitismo burgués"; allí, a un arma de la "subversión roja". Pero vuelven a brotar como hongos, y hay siempre ciertos paises en donde escapa a las persecuciones y a la hoguera. Cuando en 1954 la Conferencia General de la UNESCO adoptó la resolución

IV.1.4.422, donde se hicieron constar los resultados alcanzados por el Esperanto en el dominio de los intercambios intelectuales internacionales y el acercamiento de los pueblos del mundo, tuve mucho miedo. Por fortuna, esta resolución cayó en el olvido, al igual que el informe imparcial de la Sociedad de Naciones.

-Ya ve que no tiene por qué preocuparse.

-En pura lógica, quizá; pero mi miedo está latente y cualquier cosa lo despierta. Cuando el Cuerpo Común de Inspección estudió el problema de las lenguas... ¿Vio usted las cifras en el documento A/32/237, presentado a la Asamblea General de las Naciones Unidas el 11 de octubre último? ¡Espantoso! La conclusión "lógica" de ese informe debería haber sido la adopción por etapas de una lengua internacional, única solución que es a la vez equitativa, cómoda, económica y sin pérdida de prestigio para alguno. Tal plan fue desde luego propuesto, en privado hasta aquí, por el jefe de la delegación australiana en la ONU, el señor embajador Ralph Harry.

-Pero el Cuerpo Común de Inspección...

-...escamoteó completamente la solución evidente. ¡Uf! Sé que se puede siempre contar con la patología mental de la naturaleza humana, pero los datos del Cuerpo Común de Inspección hacían resaltar de manera tan perfecta la amplitud del problema que yo tuve pánico. Si esos señores hubieran tenido la idea de mirar lo que acaece en las sociedades esperantófonas, en donde la comunicación se hace mejor, sin injusticias, después de un tiempo de estudio limitado... pero eso no es todo.

-¿Qué sucedió todavía?

-¿Conoce usted al Director General de la UNESCO? ¿No? Un hombre encantador. Es un africano francófono. Se marchó en persona al Congreso Universal de Esperanto en Reykjavyk, el verano pasado. Pronunció un importante discurso sobre la comunicación.

-¡Qué puede importar ello!

-Importa, porque un Director General no se molesta personalmente con frecuencia para tomar la palabra en el seno de una organización no gubernamental. Importa... Observé (también estuve yo allí). Vi que él quiso saber cómo funcionaba el Esperanto entre japoneses y rusos, holandeses y brasileños, malgaches e iraníes. Escuchó al alcalde de Reykjavyk expresarse en dicha lengua, y después, al ministro de cultura de Islandia. ¡Fue monstruoso! Comprobó que la comunicación INTERNACIONAL era mucho más armoniosa en la lengua de Zamenhof que, por ejemplo, en la de Shakespeare.

-¿Es eso verdad? ¿Cómo es así?

-Ello obedece a toda clase de factores lingüísticos, fonéticos y psicológicos, pero nunca terminaría si se los detallase. No olvide que que el Esperanto fue forjado por cuatro generaciones de personas de todas las religiones, de todas las clases sociales, de todas las edades, que habitaban los más diversos paises, unidos por un solo objetivo: comunicarse. Una lengua es un organismo vivo que tiende espontáneamente hacia un equilibrio entre manejabilidad y precisión. Si se la deja evolucionar de manera natural, tiende por sí misma a alcanzar ese equilibrio.

-Pero, si lo que usted dice es cierto, ¡todas las lenguas serían igual de precisas y manejables como el Esperanto!

-No, porque no se las deja evolucionar de manera NATURAL. Todos los niños hispano-parlantes dicen espontánea-mente "he escribido" o "una tigra"; los francófonos "des chevals" o "vous disez"; todos los pequeños anglosajones utilizan formas como "foots" o "she comed".

-Es lo que hace el encanto de lenguas como el español, francés o inglés.

-Quizá, pero es lo que crea una dificultad para los extranjeros. De hecho, las lenguas son muy diversas: la belleza literaria del Esperanto, como la del chino, viene de la perfecta regularidad de sus estructuras. En la mayor parte de las lenguas nacionales, el aspecto "manejabilidad" del equilibrio natural está artificialmente inhibido por un largo condicionamiento familiar y escolar que no tiene su equivalente en Esperanto. Esta lengua nació de una abundancia de comunicaciones internacionales, y ha realizado su equilibrio a un nivel óptimo para los intercambios entre pueblos. En todo caso, el hecho está patente: entre chinos, iraníes y japoneses, por ejemplo, su SOLTURA es mucho más grande en Esperanto, después de dos años de estudio, que en inglés después de diez años.

-¿Y el Director General de la UNESCO se dio cuenta?

-Espero que no. Pero constató que personas de 60 lenguas diferentes podían reir EN EL MISMO INSTANTE las ocurrencias de uno de sus humoristas, y discutir con una animación que no se encuentra nunca en las otras asambleas internacionales en donde la barrera de las lenguas inhibe la espontaneidad.

-Anda, ¡no te preocupes! ¡Un Director General tiene tan pocos poderes! ¡Y los hombres son tan masoquistas!

-Tiene razón doctor. ¿Por qué elegirían una solución simple, agradable y democrática, cuando pueden escoger un sistema complicado y pesado, que permite a los más fuertes conservar sus privilegios? Pero el Esperanto no deja de ganar terreno. Hay actualmente, en Africa, sobre el despacho de un ministro de educación nacional, un documento proponiendo reemplazar sobre todo el continente negro la enseñanza del inglés, del francés y del portugués por el de la lengua local y del Esperanto. Todos los africanos que hubieran seguido dos años de escuela PRIMARIA podrían así comunicarse entre ellos en una lengua sin pasado colonial, mientras que actualmente, una pequeña élite sólo accede al dominio de una lengua europea, después de seis o siete años de estudios SECUNDARIOS, y eso no permite incluso a un malí el "discuter le bout de gras" con un keniata. Espero que el ministro tenga suficientemente desarrollado el sentido de sus privilegios para no comunicar dicho documento a ningún colega, pero nunca se sabe. Además...

-¿Además?

-Además temo que un día los paises cuya lengua no sea utilizada a nivel internacional pidan que se agregue el Esperanto a las lenguas de trabajo. Sería en cierto modo el representante del conjunto de lenguas menores. ¿Por qué un finlandés, un afgano, un brasileño, deben hablar la lengua de una gran potencia? Si se les ocurriera la idea...

-Pero eso no haría sino aumentar el revoltijo lingüístico, lo que sería en vuestro interés.

-No lo creo. Sería un caballo de Troya. Al cabo de algunos años, todos se darían cuenta de que el Esperanto es más manejable que las otras lenguas a nivel internacional. Y como es muy fácil de asimilar, para todos los pueblos, su uso correría el riesgo de extenderse.

-Sin contar con que ciertos "grandes" podrían utilizarlo para evitar la crítica de "imperialismo lingüístico".

-Por favor, doctor, no me agobie. No había pensado en eso. Si usted me hace pasar también por altibajos...

-Pero, a fin de cuentas, ¿por qué le da miedo que se exienda el Esperanto?

-Veamos, doctor; yo soy el diablo encargado de mantener Babel, de favorecer los malentendidos, las desigualdades sociales, las soluciones no democráticas... Si no cumplo mi cometido, mi patrón Lucifer se pondrá furioso.

-¿Qué le haría?

-No sé, me... me... me haría sufrir.

-Pero dijo que había lacanzado el máximo de sufrimiento soportable. Proponga a su patrono que le encargue otro trabajo. La comprensión lingüística nunca impidió a los hombres el destrozarse. ¿Tanto le importa Babel a su patrono?

-¿El? Quizá, en el fondo, no tanto. Es más bien a mí. Comprenda que es la obra de toda mi vida. Tiene usted razón. Desde algunos decenios, pago un precio demasiado elevado en angustia e inseguridad. Creo que voy a dimitir.

-Perfectamente. El simple hecho de tomar esa decisión va a procurarle un sosiego inmediato. Evidentemente, va a ser necesario encontrar otra actividad...

-Tengo una idea. Desde el tiempo que yo lucho contra el Esperanto, lo conozco en sus menores recovecos. Voy a proponer mis servicios como traductor.

-¿Traductor? ¡Pero, si deja de sostener Babel, no se tendrá más necesidad de traductores!

-¡Doctor, doctor, reflexione! Hay millones de páginas de archivos, obras fundamentales, reglamentos de todas clases, cuyo texto exacto en Esperanto será necesario establecer cuando la lengua se convierta en oficial. Y está la traducción literaria. Muchas obras maestras han sido publicadas en la lengua de Zamenhof, pero quedan todavía una enorme cantidad que no han sido traducidas. Además, está la idea del señor Saheb-Zamani.

-¿Qué idea?

-Viajando por el Tercer Mundo como especialista de la salud pública, el señor Saheb-Zamani constató que la formación de los técnicos y cuadros medios era muy incompleta a causa del problema de las lenguas. Traducir todos los manuales en las lenguas locales resulta muy caro. Pero aprender el inglés, el alemán o el francés suficientemente a fondo para servirse de los manuales publicados en estas lenguas no está al alcance de este personal. El señor Saheb-Zamani aprendió el Esperanto, se dió cuenta de que el Esperanto es muy exacto, o sea, muy adaptado a la enseñanza técnica, y accesible en poco tiempo a cada cual del Tercer Mundo. Entonces, propone que se organice la enseñanza del Esperanto al personal en cuestión y que se publiquen todas las obras de base en dicho idioma. Estas obras podrán tener así una tirada formidable que las haría baratas. Ciertos editores japoneses están sobre la pista de este mercado potencial. Habrá con ello para ocupar a los traductores durante años. Sabe usted; cuando se ha pasado la vida debatiéndose como un pobre diablo (si se puede decir) para impedir a los hombres dialogar y conocerse, hacer traducciones

debe ser sosegado...

-¿Lo cree usted?

-La traducción, en general, es un oficio horroroso, pero en Esperanto, es más bien escalfante. Hay dificultades, desde luego, justo suficientemente para estimular el espíritu. Pero ¡qué libertad se goza! Lo que hace difícil la traducción en las lenguas nacionales son las exigencias de la gramática, el uso, o la imposibilidad de derivar por sí mismo las palabras de que se tiene necesidad. En una lengua donde no hay ninguna dificultad arbitraria, es... es verdaderamente entusias-mante...

-Pero, mi querido amigo, ¿qué sucede? ¡Diríase que ama ese Esperanto que causa su desgracia!

-Lo amo y lo detesto. Es un malvado pilluelo. Implacablemente lógico, como los pilluelos, y desbordante de creatividad, como los mlavados pilluelos. Ma han cabreado de tal modo... Creo que le he tomado cariño. Pero, ¿qué sucede, doctor? ¡Tiene un aire soñador?

-Oh, como sabe usted, la psiquiatría es muy rentable, pero no es así siempre. Hay momentos en que sueño con otra cosa, pero sin encontrar realmente mi camino. Entonces, traductor... Traducir en francés, nunca aceptaría, pero en una lengua sin excepciones, en donde se puede dar a no importa qué palabra una forma adjetiva, substantiva, adverbial, es eso, ¿no? Traducir en una lengua fácil, donde la creatividad es siempre estimulada. Traducir incluso en horas libres o a domicilio... Dígame, querido paciente y no obstante amigo, el Esperanto, ¿dónde se aprende?

S. P. Renteaux

Artículo aparecido en la revista suiza "Dialogue", de la Organización Mundial de la Salud.

Traducido del francés por F. Zaragoza Ruiz, profesor de E.G.B. y delegado español de I.L.E.I. (Liga Internacional de Profesores Esperantistas).

Reproducido para C.U.Q. por: Marcos Cruz, Madrid, 4/8/1989

(Federación Española de Esperanto: Rodriguez San Pedro 13 3-7, 28015 Madrid, teléfono 91/4468079 (jueves, tardes); Servicio de libros de la Federación Española de Esperanto: Apartado 119, 47080 Valladolid).

OFERTAS

OFERTA

Cambio colección completa de scr-eroticas (5 pantallas)por las dos últimas pantallas de DENISE en el STRIP POKER porque me duele ya el alma de jugar a éste juegecito y no hay manera humana de ganarle a partir de la tercera pantalla (LA DEL BIKINI).

Si es necesario incluiria en la oferta una papeleta para la rifa en octubre de un ciclomotor.

Antonio Rodriguez Hernandez
Aptdo.2107
30.000 .-MURCIA

PROGRAMAS DE ESTE NUMERO

- Programa Traductor de textos en versiones Turbo (no necesita ya del toolkit II) y Qliberator.
- Programa tortuga (se carga con lrun flp1_tortuga_bot)

MANUALES TRADUCIDOS

Manual Drivers

EN PROXIMOS NUMEROS...

Un Dossier sobre el Atari ST
Versión en assembler del programa traductor de textos
El retorno (una de BBS y modem)